CA50 / UCA Client & Server

Developer & User Guide

CA50/UCA Client & Server Developer & User Guide

72E-93914-02
Revision A
November 2009

© 2008-2009 by Motorola, Inc. All rights reserved.

No part of this publication may be reproduced or used in any form, or by any electrical or mechanical means, without permission in writing from Motorola. This includes electronic or mechanical means, such as photocopying, recording, or information storage and retrieval systems. The material in this manual is subject to change without notice.

The software is provided strictly on an “as is” basis. All software, including firmware, furnished to the user is on a licensed basis. Motorola grants to the user a non-transferable and non-exclusive license to use each software or firmware program delivered hereunder (licensed program). Except as noted below, such license may not be assigned, sublicensed, or otherwise transferred by the user without prior written consent of Motorola. No right to copy a licensed program in whole or in part is granted, except as permitted under copyright law. The user shall not modify, merge, or incorporate any form or portion of a licensed program with other program material, create a derivative work from a licensed program, or use a licensed program in a network without written permission from Motorola. The user agrees to maintain Motorola’s copyright notice on the licensed programs delivered hereunder, and to include the same on any authorized copies it makes, in whole or in part. The user agrees not to decompile, disassemble, decode, or reverse engineer any licensed program delivered to the user or any portion thereof.

Motorola reserves the right to make changes to any software or product to improve reliability, function, or design.

Motorola does not assume any product liability arising out of, or in connection with, the application or use of any product, circuit, or application described herein.

No license is granted, either expressly or by implication, estoppel, or otherwise under any Motorola, Inc., intellectual property rights. An implied license only exists for equipment, circuits, and subsystems contained in Motorola products.

MOTOROLA and the Stylized M Logo and Symbol and the Symbol logo are registered in the US Patent & Trademark Office. Bluetooth is a registered trademark of Bluetooth SIG. Microsoft, Windows and ActiveSync are either registered trademarks or trademarks of Microsoft Corporation. All other product or service names are the property of their respective owners.

Motorola, Inc.
One Motorola Plaza
Holtsville, New York 11742-1300
http://www.motorola.com/enterprisemobility

Patents

This product is covered by one or more of the patents listed on the Web site:
http://www.motorola.com/enterprisemobility/patents.

Warranty

Mobile Device

The CA50 is warranted against defects in workmanship and materials for a period of one year (12 months) from date of shipment, provided the product remains unmodified and is operated under normal and proper conditions.

Accessories

CA50 accessories, such as cables and power supplies, carry a warranty term of ninety (90) days from the date of shipment.

CA50 Server Software

The CA50 Server Software ships with Motorola's standard 90-day software warranty. During the warranty period, Motorola will replace damaged or defective media. Telephone and e-mail access are also provided for the sole purpose of reporting a non-conformance issue, and for Motorola to verify the existence of the issue and determination of the conditions under which the issue may recur.

For enhanced support over the standard warranty, including full access to technical support resources and rights to download entitled software releases, Motorola recommends purchasing a CA50 Server Software Support agreement.

To maintain MSP support beyond the 90-day warranty period, Motorola recommends purchasing an MSP Software Support agreement. Note: MSP Software Support agreements must be purchased separately or in addition to CA50 Server Software Support agreements.

Revision History

Changes to the original manual are listed below:

Change	Date	Description
72E-93914-01 Rev. A	3/2008	Initial release
72E-93914-02 Rev. A	11/2009	**Updates:** UCA Server and test applications PBX support Profile Wizard Localization Scanning support Appendix H, UCA Server Backup and Restore. **Adds:** CA50 802.11a band support CA menu options Appendix I, Business Applications Supplement Appendix J, Commands and Command Usage.

Table of Contents

Patents ii
Warranty iii
Mobile Device iii
Accessories iii
CA50 Server Software iii
Revision History iv
Overview xvii

About This Guide
CA50 Configurations xviii
Chapter Descriptions xviii
Notational Conventions xx
Related Documents and Software xxi
Service Information xxi

Chapter 1: Overview
Overview 1-1
CA50 Features 1-2
Bar Code Scanning Functionality 1-2
Messaging/Paging Functionality 1-3
Walkie-Talkie (Multicast) Functionality 1-3
Voice Call (Unicast) Functionality 1-3
Components in the CA50 Solution 1-4
Quick Startup Guide Overview 1-4
Quick Startup Guide 1-5

Chapter 2: Development and Deployment Checklist
Overview 2-1
Steps in the Development and Deployment Checklist 2-1
Overview of the CA50 Solution 2-2
Step 1. UCA Server - Install 2-2

Step 2. Business Application for CA50 - Generate 2-2
Step 3. CA50 Device Settings - Optimize Settings 2-2
Step 4. UCA Server Files and Databases - Populate with Site/User Specific Information 2-3
Step 5. Wireless Infrastructure - Optimize for CA50 2-3
Step 6. IP PBX Integration (Optional) 2-3
Step 7. Deploy CA50 Solution 2-3

Chapter 3: Getting Started
Overview 3-1
Unpacking the CA50 3-1
CA50 - Front/Back Views 3-2
CA50 - Side Views 3-3
Preparing the CA50 for Use 3-4
Inserting and Removing the Battery 3-4
Battery Information 3-6
Taiwan - Recycling 3-6
Battery Facts 3-6
Battery Safety Guidelines 3-6
Charging Temperature 3-7
Average Charge Time 3-7
Charging Batteries 3-8
Charging the Battery in a CA50 3-8
Charging Spare Batteries 3-8
Charging Temperature 3-8
Battery Management Recommendations 3-9
Powering On the CA50 - Pre-deployment 3-11
CA Menu 3-12
Sleep Mode 3-13
Power Off 3-13

Chapter 4: CA50 Modes of Operation
Overview 4-1
CA50 Login - Post-deployment 4-2
Login Options 4-2
Power On 4-2
Employee Identity Validation (Scanning a Login Bar Code and entering a PIN) 4-3
Logoff Options 4-4
Display Status Icons 4-5
CA50 Keys, Actions and Features 4-7
CA50 Indications 4-10
LED Indications 4-10
Audio Indications 4-11
Screen Indications 4-12
Using an Earbud 4-13
Wireless Security Protocols for the CA50 4-14
Data Capture 4-15

Laser Scanning 4-15
Scanning Considerations 4-15
Scanning 4-16
Resetting the CA50 4-17
Performing a Warm Boot 4-17
Performing a Cold Boot 4-17
Waking the CA50 4-18
Accessing the CA Menu 4-18
Out-of-Box CA Menu Options 4-19
Walkie-Talkie (Multicast) Communication 4-31
RF Network Optimization for Walkie-Talkie Communication 4-31
Walkie-Talkie Functions 4-32
Naming Walkie-Talkie Channels 4-33
Operating Walkie-Talkie 4-34
Voice (Unicast) Calls 4-36
Communication Hierarchy 4-36
Standard (Non-PBX) Calls 4-37
PBX Calls 4-37
Voice (Unicast) Call Functionality - PBX Calls vs. Direct CA50-to-CA50 IP Based Calls 4-40
Browsing the Phone Book 4-42
Making a Voice (Unicast) Call 4-45
Receiving a Voice (Unicast) Call 4-45
Transferring a Call 4-46
Parking a Call 4-47
Retrieving a Parked Call 4-50
Park/Retrieve Troubleshooting 4-52
Voice (Unicast) Call States 4-53
Sample Business Application 4-54
Accessing the Menu and Phone Book in the Business Application 4-54
Answering an Incoming Voice (Unicast) 4-55
Launching the Sample Business Application While in a Voice (Unicast) Call 4-56
Accessing Communication Options from a Business Application 4-57

Chapter 5: UCA Client Application on the CA50
UCA Client Application Overview 5-1

Chapter 6: UCA Server
Overview 6-1
UCA Server Overview 6-2
Server Functionality 6-4
Device Registration 6-4
Login Process 6-4
User Profile Delivery 6-4
Voice Communication Setup and Maintenance 6-4
Demo Solution 6-5
Business Application and Associated Database 6-5
Development Toolkit with Sample Source Code 6-5

Required Development Tools 6-5
Theory of Operation 6-6
UCA Server Requirements and Setup 6-7
Minimum Server Hardware Requirements 6-7
Server Software Requirements. 6-7
UCA Server Installation 6-9
Installation Prerequisites 6-9
Installing UCA Server Software 6-10
Uninstalling UCA Server Software 6-42
Validate Successful Server Installation and Proper Performance 6-45
Validate Proper Launch and Login to the UCA Server Web Console 6-45
Validate a CA50 Can Log In to the UCA Server 6-46
Configuring Firewalls 6-47
System Settings 6-48
UCA Server Security 6-54
SSL (Secure Socket Layer) 6-54
Directory Level Password Protection for Sensitive Web Services 6-57
Encryption/Decryption of the UCA Server Configuration Files 6-57
Encryption of Passwords Stored in the UCA Database 6-57
CA50 Capabilities with an Offline UCA Server 6-58

Chapter 7: XML Profile for Configuration
Overview 7-1
XML Profile Overview 7-2
Delivery of the XML Profile to the CA50 7-2
XML Profile Layout 7-3
User Profile Section 7-4
User Profile Elements 7-4
Static Menus Section 7-13
Menu Elements 7-15
Changing the Ring Tone for the UCAClient Application 7-16
Modifying Phone Book Menu Behavior 7-17
Descriptions of Phone Book Web Service Methods 7-19
Menu States 7-22
Key Attribute Definitions 7-23
Setting up Walkie-Talkie Channels in the Base Profile 7-24
Base XML Profile Sample 7-25
Completed User XML Profile Sample 7-27

Chapter 8: XML Profile Builder for Device Configuration
Overview 8-1
Profile and Menu Contents 8-1
Theory of Operation 8-2
XML Profile Layout 8-3
XML Profile, Profile Section 8-3
XML Profile, Menu Section 8-3
XML Profile, Profile Section 8-4
Create New Profile 8-4

Phone Settings 8-6
Walkie-Talkie Settings 8-7
Download Files Settings 8-10
Desktop Settings 8-11
Volume Settings 8-12
Metrics & Logging 8-13
Device Settings 8-14
Client Settings 8-15
Alert Settings 8-16
Create Menus 8-21
Add Menu Elements 8-22
Add a Static Menu Element 8-23
Add a Dynamic Menu Element 8-24
Add a Client Command Menu Element 8-25
Add a Server Action Menu Element 8-26
Add Menu States 8-27
Add Menu States 8-27

Chapter 9: UCA Server WebConsole Graphical User Interface
Overview 9-1
UCA Server WebConsole Interface Overview 9-2
Login to the UCA Server System 9-3
System Status Tab View 9-5
Server Status 9-5
Devices 9-6
Phone and Walkie-Talkie Tab View 9-7
Phone Extension Status 9-7
Phone Extension Pool 9-8
Phone Extension Import File Format 9-13
Additional Phonebook Entries 9-14
Deleting Phone Extensions 9-17
Device User Accounts Tab View 9-18
Recent Users 9-18
All Users 9-19
Editing User Password 9-23
Deleting a User 9-24
Device User Import File Format 9-25
Profile Management Tab View 9-26
Create an Associated Base Profile 9-27
Editing an Associated Base Profile 9-28
Deleting Associated Base Profiles 9-29
Roles and Groups Tab View 9-30
Groups 9-30
Roles 9-33
System Management Tab View 9-37
System Settings 9-37
Console Users 9-43
Tools 9-47
UCA Server WebConsole GUI Utilities 9-52

Find Utility 9-52
Delete Utility 9-52
Import Utility 9-55

Chapter 10: CA50 Business Application Development
Overview 10-1
Business Application Overview 10-2
Business Application Development Environment 10-2
CA50 Web Application Developer Guide 10-3
Launching a Business Application 10-3
Automating the Launch of the UCA Client Application 10-3
Employee Identity Validation 10-3
Request Format 10-4
Response Format 10-4
HTML Elements 10-5
Best Practices and Considerations for CA50 Development 10-12
Data Collection 10-12
Access to Communication Options and the Web Application 10-12
Business Applications and their Extensions 10-16
Business Application Extensions Web Service 10-16

Chapter 11: UCA Server Optional APIs
Overview 11-1
UCA Server Optional APIs Overview 11-2
Location of Optional APIs and Web Messaging Service 11-3
Alternate Mechanism for Login and Authentication 11-3
Alternative Mechanism for Server Management and Maintenance of Customer Data 11-4
Mechanism for Sending Text Messages to the Device 11-4
CA50 Device User Authentication Process 11-5
Process Flow 11-5
UCA Authentication Process Diagram 11-7
Device User Authentication Web Service 11-8
Authentication Web Service Methods 11-8
Phone Extension Web Service 11-14
Phone Extension Web Service Methods 11-14
Group Management Web Service Methods 11-18
Role Management Web Service Methods 11-22
Device Messaging Web Service 11-26
Device Messaging Web Service Methods 11-26
User Management Web Service 11-29
User Management Web Service Methods 11-29
Console User Authentication Web Service 11-40
Console User Authentication Web Service Methods 11-40
Customizing the Console Login Process 11-43
UCA Server Optional API Samples 11-44
UCA Messaging Sample 11-44
UCA Phone Extension Sample 11-46

Chapter 12: Staging and Deploying CA50s

Overview 12-1
Staging and Deploying CA50s 12-2
Rapid Deployment Bar Codes 12-2
UCA.settings.xml File for Configuring the UCA Client Application 12-2
Staging Process for a CA50 12-4
Auto-launching the UCA Client Application 12-6
Staging and Localization 12-6
OS/UCA Client Application Upgrades Pushed from MSP Provision 12-7
Minimizing Network Traffic from the MSP Agent 12-7

Chapter 13: Site Installation Guidelines

Overview 13-1
Site Installation Guidelines Overview 13-2
Maximum Number of CA50s Deployed Per Site 13-2
Wireless Network Optimization 13-2
Decrease Roam Times Through a Channel Mask 13-3
Battery Management 13-6
UCA Server Requirements 13-7
Phone Information 13-7
Operations Information 13-7
Walkie-Talkie Information 13-7
IT Information 13-7
Stages of Planning and Preparation 13-8
Pre-deployment Activities - First Site 13-8
Deployment Activities - Per Site 13-8
Planning and Preparation Activities 13-9
Developing a List of Roles 13-9
Developing a List of Groups 13-9
Planning Walkie-Talkie Channels and Addresses 13-9
Planning User Authentication (Device Users and Console Users) 13-9
Planning PBX Integration 13-10
Developing User Profile Templates and Associations 13-10
Developing Custom Business Applications (optional) 13-10
Identifying Site Specific Settings 13-10
Configuring Site User Lists with Role/Group Assignments 13-10
Configuring a Site Phone Extension Pool 13-10
Configuring Additional Phonebook Entries 13-10

Chapter 14: Maintenance, Troubleshooting and Error Messages

Overview 14-1
Maintaining the UCA Server 14-2
Scheduled Data Maintenance 14-2
Keeping System Data and Configuration Up to Date 14-3
Maintaining the CA50 14-3
Cleaning and/or Disinfecting 14-3
Troubleshooting 14-4
UCA Server Installation/Uninstall 14-4

CA50 Client 14-6
CA50 Device 14-11
UCA Server System Error Code Number Index 14-13
Client Generated Error Messages 14-13
UCA Server System Error Code Number Ranges 14-16
General Errors 14-16
UCA Database Errors 14-17
General Errors 14-18
UCA Database Errors 14-34

Chapter 15: Localization
Overview 15-1
UCA Server Localization 15-2
CA50 Device Localization 15-3
Business Application Localization 15-3
Installation 15-4

Appendix A: Specifications
Technical Specifications A-1
CA50 Accessory Specifications A-4

Appendix B: Accessories
Overview B-1
Accessories Summary B-2
Eight-Slot CA50 Charging Cradle B-3
Setup B-3
Battery Charging Indications B-4
Charging Temperature B-4
RF Connectivity B-4
Single-Slot CA50 Charging Cradle with Spare Battery Charger B-5
Setup B-5
Charging the Battery B-6
Charging the Spare Battery B-6
Battery Charging Indications B-7
Charging Temperature B-7
RF Connectivity B-7
Eight-Slot Spare Battery Charger B-8
Spare Battery Charging B-8
Battery Charging Indications B-9
Charging Temperature B-9
LED Indications B-9
Earbud B-10
Lanyard with Clip B-11
Attaching and Removing the Lanyard B-11
USB Cable B-12
Communication Setup B-12
Wall Mount Brackets B-13

KT-102376-01R Bracket B-13
KT-102375-01R Bracket B-15

Appendix C: Enabling/Disabling Scanner Symbologies
Overview C-1
Enabling/Disabling Symbologies C-1
Supported Code Types C-2
Enabled Code Types C-2
Configuring Symbologies C-3
Configuring Symbologies During the Login Process C-12
Configuring Symbologies in a Business Application C-12

Appendix D: Deployment Example - UCA Server Install & Basic User Setup
Overview D-1
Outline of Steps to Perform D-2
Step 1. Required Site Settings for this Exercise D-3
Wireless Network Site Settings D-3
CA50 and UCA Server Settings D-3
Business Application Settings D-3
Step 2. Setup Static IP on Server Hardware D-4
Step 3. Install UCA Server Software D-4
Step 4. Validate Successful Server Installation on Laptop D-5
Step 5. Create an XML Profile Called "Footwear.xml" D-6
Step 6. Populate the UCA Server Database with 2 Users D-11
Step 7. Generate Employee Login Bar Codes D-15
Step 8. RF Network D-16
Step 9. Use MSP 3.1 to Prepare CA50 Devices for Use D-16
Create RD Bar Codes for the CA50 ..... D-17
Step 10. Deploy CA50s D-18
Deploy CA50s at the Site Location D-18
Validate that the CA50 was Deployed Properly D-18
Employee Logs in to the UCA Server D-18

Appendix E: Self Contained Pre-Built Demo
Overview E-1
Outline of Steps to Perform E-2
Step 1. Hardware Required to Run the Demo Solution E-2
Demo Solution System Diagram E-3
Step 2. Setup Static IP on Server Hardware E-4
Step 3. Install UCA ServerDemo Software E-4
Step 4. Validate Successful Server Installation on Laptop E-5
Step 5. Configure the RF Hardware E-5
Step 6: Scan Rapid Deployment Bar Codes to Deploy CA50s E-11
Step 7: Scan Employee Login Bar Code E-14
Managers E-14
Sales Associates E-14
Front End Associates E-16

Step 8: Ready to Demonstrate the CA50 Voice and Scanning Functionality E-17
Price Lookup SKUs for the Demo Solution E-17
Step 9. Modifying the Demo Solution: Creating New Users E-18

Appendix F: Rapid Deployment
Overview F-1
Accessing the Rapid Deployment Client F-1
Scanning Rapid Deployment Bar Codes F-1
Rapid Deployment Window Options F-3
Errors F-3

Appendix G: ActiveSync
Overview G-1
Installing ActiveSync G-2
Setting Up an ActiveSync Connection G-2
Connecting to the CA50 ..... G-3
Downloading Data using an ActiveSync Connection G-3

Appendix H: UCA Server Backup and Restore
UCA Backup & Restore H-1
Key Features H-1
Provide Database Credentials H-2
Backup the UCA Server H-2
Restore the UCA Server H-3
Reconfigure the UCA Server H-4

Appendix I: Business Applications Supplement
Introduction I-1
Adding a Menu Option I-2
Launching a Business Application on UCAClient Startup I-3
Setting the HTML Page Title I-4
Adding a Timer to Visible and Vibrate Indicators I-4
Enabling the BackLight for HTML Pages I-5
Download Files / Delete Downloaded Files I-5
Download Files to the CA50 Device Upon User Login I-5
Download Files to the CA50 Device After User Login I-6
Delete Downloaded Files I-6
Adding User Information to the HTTP Header I-6
Assigning a Default Value to the SpinBoxControl I-7
Example I-7
Code Samples for HTML Pages in Business Applications I-7
Enlarging Fonts in a Menu List I-7
Entering a Date in the Spin Box (HTML) I-8
Scanning Item/Enter Quantity Using Spin Boxes I-9
Adding and Enlarging Images/Fonts I-10
Enlarging Special Characters I-12

LED Settings (HTML Business Application) I-13

Appendix J: Commands and Command Usage

Introduction J-1
Commands Usage in Profiles J-2
Commands Usage in Business Applications (HTML Pages) J-4
Commands Usage in Notification Messages J-14

Index

Tell Us What You Think...

Quick Startup

About This Guide

Overview

This guide provides information for developers and integrators who are writing, deploying and maintaining applications for the UCA server and client.

NOTE Screens and windows pictured in this guide are samples and can differ from actual screens.

NOTE 1. For a complete checklist outlining all the development and deployment activities required to implement a CA50 solution, see the *Development and Deployment Checklist on page 2-1*.

2. For a table listing the most commonly requested topics within this manual, see the *Quick Startup Guide on page 1-5* and *Quick Startup Guide* on the last page of the manual.

CA50 Configurations

Table 1-1 *CA50 Configurations*

Model	Features
CA5090-0U0LF5KV11R	1-pack
	802.11b/g/a radio Note: Upon initial release, the CA50 hardware is WMM compatible but not software enabled/supported. A future software release will enable WMM support.
CA5090-0U0LF5KV18R	8-pack
	802.11b/g/a radio Note: Upon initial release, the CA50 hardware is WMM compatible but not software enabled/supported. A future software release will enable WMM support.

Chapter Descriptions

Table 1-2 lists the topics covered in this guide.

Table 1-2 *Guide to Chapters*

Chapter	Description
Chapter 1, Overview	Provides a general overview of UCA server functionality and CA50 operation, components and features.
Chapter 2, Development and Deployment Checklist	Identifies all the significant steps required to generate and deploy a CA50 solution in a production environment.
Chapter 3, Getting Started	Provides information about installing/removing the battery, charging the battery, sleep mode and powering the CA50 on.
Chapter 4, CA50 Modes of Operation	Provides information about using the CA50, post-deployment.
Chapter 5, UCA Client Application on the CA50	Provides information about how to set up the CA50 client to connect and communicate with the UCA server.
Chapter 6, UCA Server	Provides information about the UCA server software (core server components, Web based console for server setup and maintenance, Demo Solution and Development and Integration Tool Kit) and the UCA server database, and includes a theory of operation.
Chapter 7, XML Profile for Configuration	Defines what a profile is and what settings are included in the profile. In addition, the manual outlines how to develop and customize a profile for a CA50.

Table 1-2 *Guide to Chapters (Continued)*

Chapter	Description
Chapter 8, XML Profile Builder for Device Configuration	Provides information about the Profile Builder application which provides a user interface for an administrator to create new UCA XML profiles and modify existing UCA XML profiles.
Chapter 9, UCA Server WebConsole Graphical User Interface	Provides information about the UCA server WebConsole GUI and how to use it for UCA server configuration and setup.
Chapter 10, CA50 Business Application Development	Provides guidance for developing Web applications that reside on a Web server and are accessed by the CA50 UCA client application.
Chapter 11, UCA Server Optional APIs	Outlines how to use the optional APIs that are installed with the UCA server. These APIs are utilized to extend the standard Motorola provided UCA server functionality. In addition, the APIs enable integration of the server into the existing store processes.
Chapter 12, Staging and Deploying CA50s	Describes client/server features that allow packaging applications and procedures for deploying applications onto the CA50 client. Describes how to program the CA50 to automatically launch directly into the UCA client application with no user intervention.
Chapter 13, Site Installation Guidelines	Provides information for planning and preparing the UCA server and CA50 client for proper customer deployment. This includes planning, preparing and maintaining a customer site environment to organize phone book groups, identify phone extensions, walkie-talkie channels, etc.
Chapter 14, Maintenance, Troubleshooting and Error Messages	Provides instructions on cleaning and storing the device, and provides troubleshooting solutions for potential problems during device operation.
Chapter 15, Localization	Provides information about the functionality of localized versions of the UCA client software on the CA50 device.
Appendix A, Specifications	Provides technical specifications for the CA50 client and cradles.
Appendix B, Accessories	Provides descriptions and features of the CA50 accessories.
Appendix C, Enabling/Disabling Scanner Symbologies	Provides information for enabling and disabling symbologies on the CA50.
Appendix D, Deployment Example - UCA Server Install & Basic User Setup	A step-by-step exercise for the installation of a UCA server, its configuration and the deployment of CA50s.

Table 1-2 *Guide to Chapters (Continued)*

Chapter	Description
Appendix E, Self Contained Pre-Built Demo	Provides information about the CA50 Demo Solution which includes all the documentation and bar codes required to configure and demonstrate walkie-talkie communication, voice calls and price lookup functionality.
Appendix F, Rapid Deployment	Provides information about using Rapid Deployment bar codes to download software to CA50s.
Appendix G, ActiveSync	Provides information about using Microsoft ActiveSync to manually update data on the CA50.
Appendix H, UCA Server Backup and Restore	Provides information about the UCA server backup and restore process.
Appendix I, Business Applications Supplement	Provides examples and tips for writing business applications for the CA50 device.
Appendix J, Commands and Command Usage	Provides the commands used in the profile, business applications and notification messages that can be executed on the CA50 device

Notational Conventions

The following conventions are used in this document:

- The terms client, device and terminal refer to the CA50 hand-held device.
- *Italics* are used to highlight the following:
  - chapters and sections in this and related documents
  - dialog box, window, and screen names
  - drop-down list and list box names
  - check box and radio button names
  - icons on a screen
- **Bold** text is used to highlight the following:
  - key names on a keypad
  - button names on a screen
- Bullets (•) indicate:
  - action items
  - lists of alternatives
  - lists of required steps that are not necessarily sequential
- Sequential lists (e.g., those that describe step-by-step procedures) appear as numbered lists
- Text boxes:

NOTE This symbol indicates something of special interest or importance to the reader. Failure to read the note does not result in physical harm to the reader, equipment or data.

CAUTION This symbol indicates that if this information is ignored, the possibility of data or material damage may occur.

WARNING! **This symbol indicates that if this information is ignored, the possibility that serious personal injury may occur.**

IMPORTANT This symbol indicates that an important step is required to complete a task correctly.

Related Documents and Software

The following documents provide more information about the CA50.

- *CA50 Quick Start Guide*, p/n 72-71770-xx
- *CA50 Regulatory Guide*, p/n 72-96711-xx
- *CA50 Cradles Regulatory Guide*, p/n 72--96712-xx
- Latest ActiveSync software, available at: http://www.microsoft.com.

For the latest version of this guide and all guides, go to: http://www.motorola.com/enterprisemobility/manuals.

Service Information

If you have a problem with your equipment, contact Motorola Enterprise Mobility support for your region. Contact information is available at: http://www.motorola.com/enterprisemobility/contactsupport.

When contacting Enterprise Mobility support, please have the following information available:

- Serial number of the unit
- Model number or product name
- Software type and version number

Motorola responds to calls by e-mail, telephone or fax within the time limits set forth in service agreements.

If your problem cannot be solved by Motorola Enterprise Mobility Support, you may need to return your equipment for servicing and will be given specific directions. Motorola is not responsible for any damages incurred during shipment if the approved shipping container is not used. Shipping the units improperly can possibly void the warranty.

If you purchased your Enterprise Mobility Solutions product from a Motorola business partner, please contact that business partner for support.

(Page intentionally blank.)

Chapter 1 Overview

Overview

The CA50 is a compact mobile device for in-building use that combines bar code scan capability with WLAN voice and data communications. It offers mobile phone-style voice communications (walkie-talkie and voice calls), wireless bar code scanning, messaging/paging and data application support for information retrieval. CA50 voice capabilities greatly improve the accessibility of your workforce - mobile workers are never more than a phone call away. The built-in bar code scanner and display, allow the CA50 to be used for a virtually unlimited number of business applications such as price lookup.

The CA50 is easy to use and increases worker productivity and availability, while reducing errors and improving customer service and satisfaction. The familiar and intuitive mobile phone-style user experience virtually eliminates the need for training. Small and light enough to wear on a lanyard or apron, or slip in a pocket, this device is the one device all your mobile associates should carry.

The CA50 can be deployed either as a personal or shared device. In environments where workers change shifts and days, enterprises can enable employees to select any available CA50 - and a simple log in configures the device on the fly with the right profile for that associate.

In addition to the CA50 mobile device, the CA50 solution includes a UCA server. The UCA server software uses a database to store the data for configuring, maintaining and tracking the CA50s.

Other sections in this chapter include:

- A list of the components in the CA50 system. See *Components in the CA50 Solution on page 1-4*.
- A list of CA50 functionality. See *CA50 Features on page 1-2*.
- A table listing the most commonly requested topics within this manual. See the *Quick Startup Guide on page 1-5* and on the last page of the manual.

✓ ***NOTE*** For a complete checklist outlining all the development and deployment activities required to implement a CA50 solution, see the *Development and Deployment Checklist on page 2-1*.

CA50 Features

The CA50 is a Voice-over-IP (VoIP) enabled wireless scanner. It converges voice, bar code scanning and data capabilities into a single device.

Table 1-1 lists CA50 features.

Table 1-1 *CA50 Features*

Feature	Benefit
Wireless Bar Code Scanning	Enables productivity-enhancing on-the-spot applications such as price verification and inventory check
Wireless Messaging/Paging	Enables more rapid response to customer request for assistance; enables better management of employees through wireless distribution of tasks and real time acknowledgement of completion
WLAN-based Walkie-Talkie Communications	Instant employee accessibility through a call to a group of workers simultaneously improves productivity and customer service; elimination of annoying pages and associated long response times
WLAN Voice Calls from CA50 to CA50	Enables employees to place or receive a call anywhere in the enterprise to or from another CA50 carrying associate; improved accessibility for customers and managers; ability to obtain needed information more rapidly for on-the-spot decisions
WLAN Voice Calls through an IP PBX	PBX integration allows inbound customer calls to be forwarded directly to the right employee in the right department. Also associates can call traditional land line phones through the PBX.

NOTE CA5090-0U0LF5KV11R 1-pack 802.11b/g/a and CA5090-0U0LF5KV18R 8-pack 802.11b/g/a radios

Bar Code Scanning Functionality

Third party business applications can utilize the CA50 1-Dimensional bar code scanner to scan bar codes such as UPC. An example of a third party business application is price lookup.

For additional information on business applications see *Chapter 10, CA50 Business Application Development*.

Messaging/Paging Functionality

CA50 development and integration tools offer an optional server API that provides methods by which an application developer can generate various types of *messages* that can be sent to a CA50 from the server.

- Text message - the device displays text to the user.
- Text message with device based audio - the device displays text and plays the specified audio file (.wav file preloaded onto the device).

For additional information on messaging see pages *11-26* and *11-44*.

Walkie-Talkie (Multicast) Functionality

Walkie-talkie communication is a one-to-many (multicast), half duplex, group communication mode that utilizes the wireless infrastructure of the facility to communicate within the four walls of the facility.

For example, an associate carrying a CA50 can press the **Walkie-Talkie** key and instantaneously communicate to a group associates carrying CA50s. While the first associate talks, the remaining associates carrying CA50s cannot respond until the first associate releases the **Walkie-Talkie** key.

For additional information on walkie-talkie communications see *page 4-31*.

Voice Call (Unicast) Functionality

A voice (unicast) call is a one-to-one full duplex communication mode that utilizes the wireless infrastructure of the facility to communicate within the four walls of the facility.

For example, an associate carrying a CA50 can select another associate to call from a phone book and carry on a full duplex, two way conversation with the second associate.

When a CA50 communicates via a voice call without a PBX, there is support for the following voice call functionality:

- Initiating a voice call to another CA50 via a UCA server dynamically managed phone book
- Receiving a voice call from another CA50
- Placing a voice call on hold and then resuming the call (off hold)
- Transferring a voice call to another extension (in the absence of PBX, call transfer is supported only in SIP mode)

  For additional information on voice calls see *page 4-36*.

If an IP PBX is integrated into the CA50 system, there is support for the following additional features:

- Initiating a voice call to a traditional land line phone via a UCA server dynamically managed phone book
- Receiving a switch board routed phone call from a customer
- Receiving a voice call from an in-building land line
- Transferring a voice call to another extension or a PSTN number via PBX using dynamically managed phone book. For additional information on IP PBX see *page 4-36*
- Parking a voice call on the PBX for pick up by another user using CA50 menu options

For additional information on IP PBX see *page 4-36*.

Components in the CA50 Solution

The following components are required to implement the CA50 solution:

- Mobile device: CA5090
  - UCA client application (for more details see *page 5-1*)
- Software:
  - UCA server software (for more details see *page 6-7*)
  - MSP 3.x software (stage and provision provided free with the purchase of UCA server software)
- Charging accessories
- Wireless network (sold separately from components in the CA50 solution)

The following components are optional to implement the CA50 solution:

- IP PBX (optional - not sold by Motorola)
- Accessories - various, such as spare batteries

Quick Startup Guide Overview

For a table listing the most commonly requested topics in this manual, see the *Quick Startup Guide on page 1-5* and the back cover of the manual.

NOTE The *Quick Startup Guide on page 1-5* is designed to be removed from the manual for easy reference.

Quick Startup Guide

Introduction to CA50 Solution 1-1, 1-2

Walkie-Talkie 3-2, **4-31**, 4-34, 4-56, 5-1, 6-4, 9-7, 13-7, 14-37, E-1

- Channel Selection 4-34

Voice Calls 5-1, 6-4, 9-14, 9-7, 14-18, 14-17, **4-36**, E-1

- Phone Book 13-7, 13-10, 4-36, **4-42**
- Making/Receiving/Transferring Calls **4-45**, 4-46

Business Applications Like Price Lookup 4-54, 6-5, **11-2**, 13-10, E-1

Messaging/Paging **1-2**, 11-26, 11-44

Components in CA50 Solution 1-4

A Good Place to Start 1-1, 1-2

Development and Deployment Checklist 2-1, **2-2**, 13-2

Self Contained Pre-Built Demo E-1

Deployment Example: UCA Server Install and Basic User Setup D-1

CA50 - Hardware

Physical Key Layout **3-2**, 4-7, 4-19

Volume Control/Mute Modes/Headset 3-2, 4-7, 4-13

CA50 - Configuration/Staging/Deploying

Reboot Device/Access the CA Menu 3-12, 4-17, **4-18**

Device Staging and Deployment using MSP 3-12, 4-19, **12-1**, 12-2, D-16, F-1

XML Profile Configures CA50 when Downloaded from Server **7-1**, D-6

Battery Management Recommendations 3-6, **3-9**, 4-4, A-1, B-8, B-12

Determining Version Information (OS, Client, etc.) 4-2, **4-29**

Server 5-1, **6-2**, D-1

Installing/Uninstalling Server Software **6-10**, 6-42, 6-45

- Server Hardware / Software Requirements 6-7
- Validate Successful Server Installation and Proper Performance 6-45

Web Console for Server Setup and Maintenance 9-2, 14-2

- Login and Password **9-4**, 6-13
- Populate Database with User/Site Specific Information 2-3
- CA50 Capabilities with UCA Server 6-58
- Importing Users and Extensions **9-13**, 9-25
- Troubleshooting/Error Codes **14-4**, 14-6, 14-18
- Alternate Administrative Mechanism to Manage Customer Data **11-5**, 11-14, 11-18, 11-22, 11-29, 11-36

Business Applications 5-1, **10-2**, 10-2, 10-12, E-1

Launching Business Applications 10-2, 12-2

Employee Identity Validation: Login Bar Codes and PIN **4-3**, 11-5, 11-40, D-15, E-14

HTML Elements 10-5

Troubleshooting/Error Codes **14-4**, 14-6, 14-13

PBX Integration **4-36**, 4-38, 6-48, 9-13, 9-41, 13-10

(Page intentionally blank.)

Chapter 2 Development and Deployment Checklist

Overview

The Development and Deployment Checklist identifies all the significant steps required to generate and deploy a CA50 solution in a production environment.

To enable easy demonstration of the CA50, a self-contained pre-build Demo Solution is included on the UCA server Installshield. For details see*page E-1*.

Steps in the Development and Deployment Checklist

Table 2-1 *Checklist - Server Development and CA50 Deployment Steps*

Step	Checklist for Server Development and CA50 Deployment	Application Developer	Integrator	Demo Solution
1	UCA Server - Install	✓	✓	✓
2	Business Application for CA50 - Generate (optional)	✓	NA	NA[1]
3	CA50 Device Settings - Optimize Settings	✓	NA	NA[2]
4	UCA Server Files and Databases - Populate with Site/User Specific Info	✓	✓	NA[1]
5	Wireless Infrastructure - Optimize for CA50	✓	✓	NA[1]
6	IP PBX Integration (Optional)	Optional	Optional	NA[1]
7	CA50s - Site Deployment	✓	✓	NA[2]

✓ ***NOTE*** When using the Demo Solution, no user development is required for this step.

✓ ***NOTE*** To simplify CA50 deployment, bar codes to configure the Ca50 are included in *Appendix E, Self Contained Pre-Built Demo*.

Overview of the CA50 Solution

- Overview of the System 1-1
- Components in the CA50 Solution 1-4
- Walkie-Talkie 1-3, 4-31
- Voice Calls 1-3, 4-36
- Business Applications like Price Check 1-2, 10-2
- Messaging / Paging 1-3, 11-26, 11-44
- Self Contained Pre-build Demo E-1

Step 1. UCA Server - Install

- Overview of Server Functionality 5-1, 6-2
- Server Hardware Requirements 6-7
- Server OS Requirements 6-7
- Install UCA Server Software 6-9
- Validate UCA Server Software Installed Properly . 6-45

Step 2. Business Application for CA50 - Generate

- Definition of a Business Application 10-2
- Build Web Business Application 10-3
- Launching Business Application 10-3, 12-2
- Employee Identity Validation 4-3, 10-3, 11-5, 11-40, E-14
- Reference Implementation: (Demo Solution) E-1

Step 3. CA50 Device Settings - Optimize Settings

- Rapid Deployment Bar Codes 12-1, F-1
- XML Profiles for Configuring the CA50 . . . . . 7-1, D-6
- Optimize Battery Life 3-9

Step 4. UCA Server Files and Databases - Populate with Site/User Specific Information

- Create Users Accounts *9-20*, *D-1*
  - Importing User Information *9-25*
- Create Phone Book Extensions *9-9*, *D-1*
  - Importing Extensions *9-13*, *9-25*
- Create XML Profiles *7-1*, *D-1*
- Create Groups and Roles. *9-30*, *D-1*
- (Optional) IP PBX Integration *9-41*, *D-1*
- Self Contained Pre-built Demo *E-1*
- Deployment Example *D-1*, *D-11*

Step 5. Wireless Infrastructure - Optimize for CA50

- Generate Wireless Profile (*page 3-9*)
- Use the CA50 to perform 1st RF Site Survey

  Note: The CA50 must be used for RF site surveys that require optimization of CA50 voice communications. For these sites the CA50 is the only acceptable survey tool. Other mobile devices (such as an MC50) do not produce an equivalent signal strength indication as displayed on the CA50.
  - Ping (*page 4-27*)
  - Signal Strength Meter on the CA50 (*page 4-22*)
- Optimize RF Infrastructure
- Use the CA50 to perform 2nd, validation RF Site Survey (THIS is a mandatory step).
- Recommended Signal Strength (*page 3-10*)

Step 6. IP PBX Integration (Optional)

- IP PBX Integration to UCA Server *4-37*, *6-48*, *9-13*, *9-41*

Step 7. Deploy CA50 Solution

- Generate Rapid Deployment Bar Codes *12-1*
- Charge CA50 Device *3-4*, *3-8*
- Deploy CA50s at Store Location
  - Boot to CA Menu *3-11*, *3-12*, *4-17*, *4-18*, *4-19*
  - Launch Rapid Deployment Client on CA50 . . . . *F-1*, *4-19*
  - Scan Rapid Deployment Bar Codes *F-1*
- Validate the CA50 was Deployed Properly
  - Confirm Network Connectivity *4-5*
- Scan Bar Code to Login to UCA Server *4-3*
- Train Employees to Use the CA50 . . . . . *1-1*, *1-2*, *3-2*, *4-3*, *4-7*, *4-31*, *4-36*, *E-17*

(Page intentionally blank.)

Chapter 3 Getting Started

Overview

This chapter shows the parts of the CA50 and provides information about installing/removing the battery, charging the battery, sleep mode and powering the CA50 on.

Valuable topics in this chapter include:

- CA50 views (*page 3-2*)
- Inserting and removing the battery (*page 3-4*)
- Average charge time (*page 3-7*)
- Charging batteries (*page 3-8*)
- Battery management recommendations (*page 3-9*)
- Powering on the CA50, pre-deployment (*page 3-11*)
- CA menu (*page 3-12*)
- Sleep mode (*page 3-13*)
- Powering off the CA50 (*page 3-13*).

NOTE 1. For a complete checklist outlining all the development and deployment activities required to implement a CA50 solution, see the *Development and Deployment Checklist on page 2-1*.

2. For a table listing the most commonly requested topics within this manual, see the Quick Startup Guide on the last page of the manual.

Unpacking the CA50

Carefully remove all protective material from the CA50 and save the shipping container for later storage and shipping. Verify that you received the following equipment:

- CA50
- Lithium-ion battery
- Regulatory Guide
- Quick Start Guide.

Inspect the equipment. If any equipment is missing or damaged, contact the Motorola Enterprise Mobility support center immediately. See *Service Information on page xxi* for contact information.

CA50 - Front/Back Views

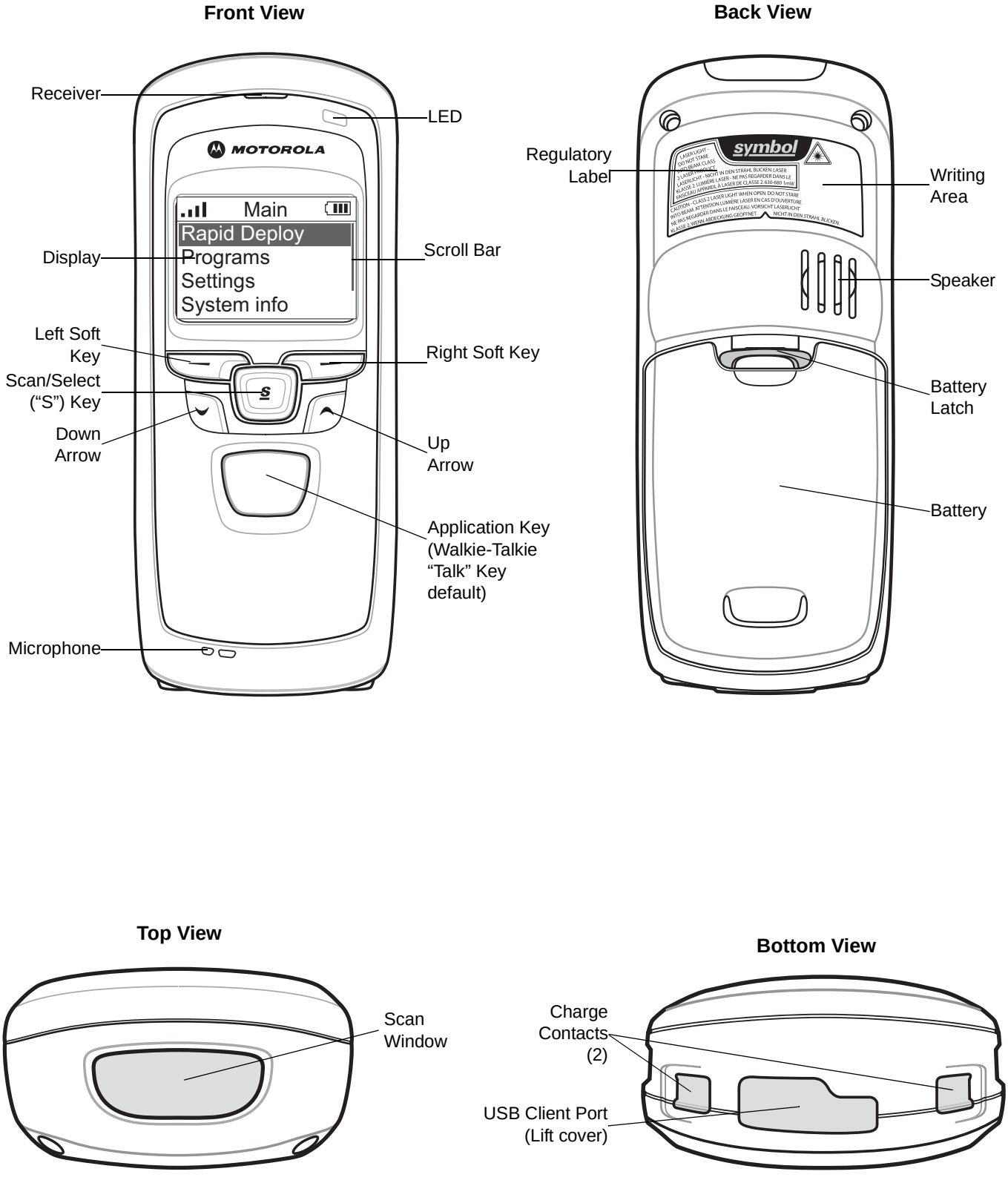


Figure 3-1 *CA50 - Front/Back Views*

CA50 - Side Views

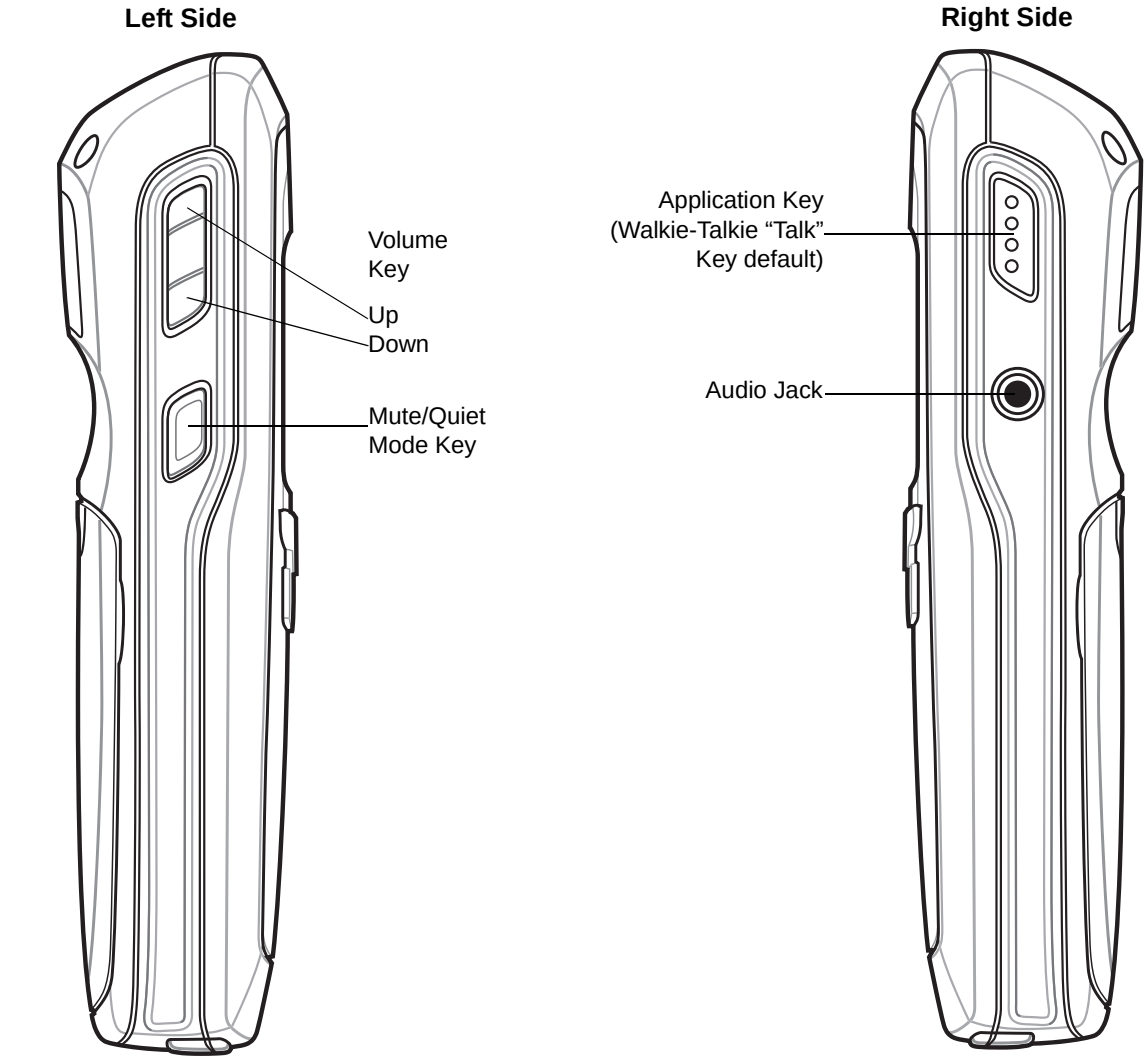


Figure 3-2 *CA50 - Side Views*

Preparing the CA50 for Use

To prepare the CA50 for first time use:

- Install the battery.
- Charge the CA50 (charge spare batteries as needed).
- Configure the CA50 device via Rapid Deployment bar codes (see *Appendix F, Rapid Deployment*).

NOTE When Rapid Deployment completes device configuration, the device has a valid network configuration and the UCA client application communicates with a UCA server (assuming a valid UCA server address is contained within the Rapid Deployment bar codes).

RF settings and settings to communicate to the UCA server can be set several ways: by scanning one set of bar codes that configure both RF and UCA server settings; by scanning two separate sets of bar codes, one containing RF settings, the other containing UCA server settings; or, by scanning one set of RF parameters and setting the UCA server through an On Demand server function of MSP.

To completely update a CA50 device, up to three sets of Rapid Deployment bar codes may need scanning. When Rapid Development is complete, cold boot the CA50 to boot up into the UCA client application at which time the device communicates to the server and receives the login screen.

Charge the battery before or after it is installed. Use one of the spare battery chargers to charge the battery (out of the CA50), or one of the CA50 device chargers to charge the battery installed in the CA50.

Inserting and Removing the Battery

Inserting the Battery

Before using the CA50, insert the lithium-ion battery provided with the device.

1. Insert the battery, bottom first, into the battery compartment in the back of the device.
2. Press the battery down into the battery compartment until the battery release latch snaps into place.

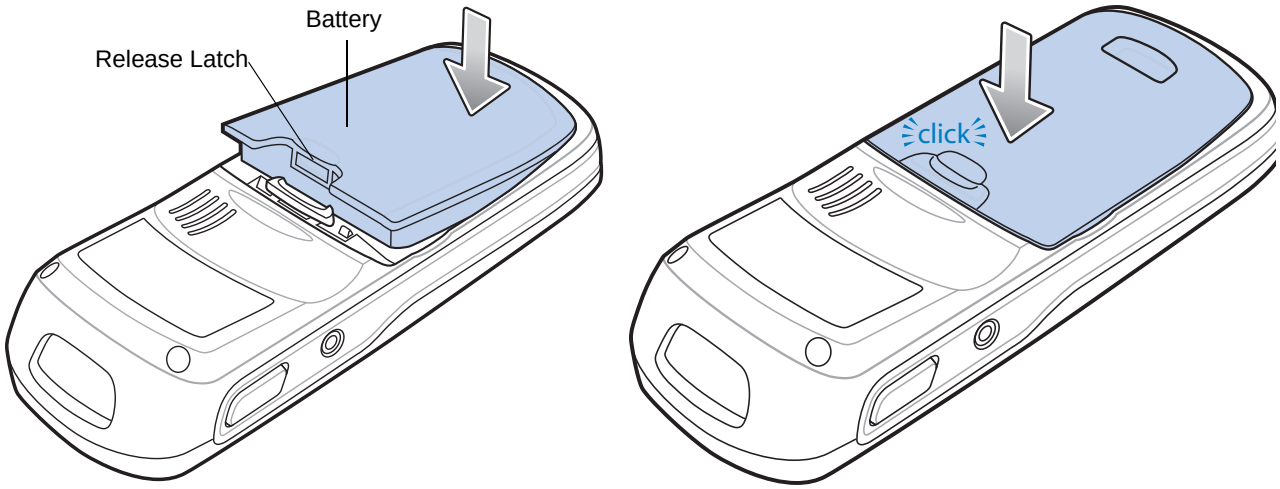


Figure 3-3 *Inserting the Battery*

NOTE Position the battery correctly, with the battery charging contacts pointing towards the bottom of the CA50.

Removing the Battery

1. Hold the device on its side and lift the release latch to disengage the battery from the device.

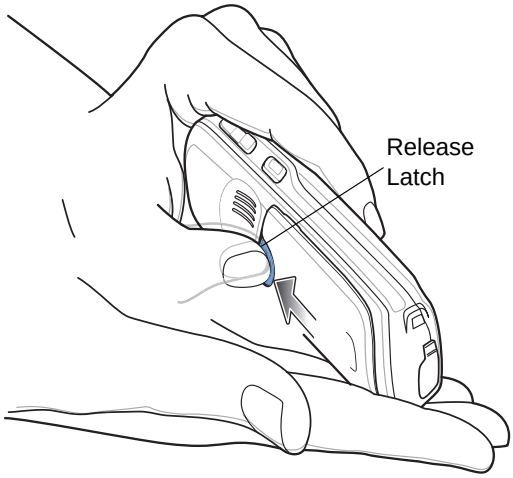


Figure 3-4 *Lift Release Latch*

2. Insert a finger nail into the indent on the bottom of the battery and push downward. The battery should eject.

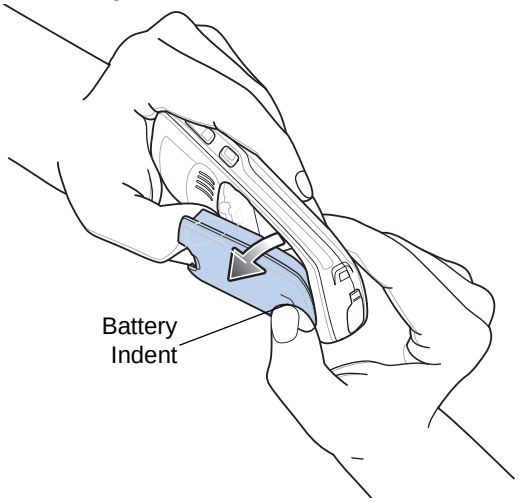


Figure 3-5 *Push Battery Indent*

NOTE In the event of a tight battery fit, the battery may not drop out of the well as pictured in *Figure 3-5*. Should this occur, releasing the battery latch adequately and pushing the battery indent lifts the battery slightly out of the well so that it can be gripped and pulled out.

Battery Information

Only genuine Motorola batteries (p/n 82-83172-01) should be used with CA50 devices and charging accessories.

Taiwan - Recycling

EPA (Environmental Protection Administration) requires dry battery producing or importing firms in accordance with Article 15 of the Waste Disposal Act are required to indicate the recycling marks on the batteries used in sales, giveaway or promotion. Contact a qualified Taiwanese recycler for proper battery disposal.

Battery Facts

Motorola rechargeable battery packs are designed and constructed to the highest standards within the industry. However, there are limitations to how long a battery can operate or be stored before needing replacement. Many factors affect the actual life cycle of a battery pack, such as heat, cold, harsh environmental conditions and severe drops.

When batteries are stored over six (6) months, some irreversible deterioration in overall battery quality may occur. Store batteries at half of full charge in a dry, cool place, removed from the equipment to prevent loss of capacity, rusting of metallic parts and electrolyte leakage. When storing batteries for one year or longer, the charge level should be verified at least once a year and charged to half of full charge.

Replace the battery when a significant loss of run time is detected. Standard warranty period for all Motorola batteries is 30 days, regardless if the battery was purchased separately or included as part of the mobile computer. For more information on Symbol batteries, please visit: http:/mysymbolcare.symbol.com/battery/batbasics1.html.

Battery Safety Guidelines

- The area in which the units are charged should be clear of debris and combustible materials or chemicals. Particular care should be taken where the device is charged in a non commercial environment.
- Follow battery usage, storage, and charging guidelines found in the user's guide.
- Improper battery use may result in a fire, explosion, or other hazard.
- To charge the mobile device battery, the battery and charger temperatures must be between +32º F and +95º F (0º C and +35º C)
- Do not use incompatible batteries and chargers. Use of an incompatible battery or charger may present a risk of fire, explosion, leakage, or other hazard. If you have any questions about the compatibility of a battery or a charger, contact Motorola Enterprise Mobility support.
- Do not disassemble or open, crush, bend or deform, puncture, or shred.
- Severe impact from dropping any battery-operated device on a hard surface could cause the battery to overheat.
- Do not short circuit a battery or allow metallic or conductive objects to contact the battery terminals.
- Do not modify or remanufacture, attempt to insert foreign objects into the battery, immerse or expose to water or other liquids, or expose to fire, explosion, or other hazard.

Do not leave or store the equipment in or near areas that might get very hot, such as in a parked vehicle or near a radiator or other heat source. Do not place battery into a microwave oven or dryer.

- Battery usage by children should be supervised.
- Please follow local regulations to promptly dispose of used re-chargeable batteries.
- Do not dispose of batteries in fire.
- Seek medical advice immediately if a battery has been swallowed.
- In the event of a battery leak, do not allow the liquid to come in contact with the skin or eyes. If contact has been made, wash the affected area with large amounts of water and seek medical advice.
- If you suspect damage to your equipment or battery, contact Motorola Enterprise Mobility support to arrange for inspection.

Charging Temperature

IMPORTANT:

Charge batteries in temperatures from 0°C to 35°C (32°F to 95°F).

Note that at temperatures above 30°C the charging temperature is monitored and controlled by the device and the charging accessory. Charging is halted at temperatures above 35°C.

The device or accessory indicates when charging is disabled due to abnormal temperatures via its LED and/or battery icon. See *Table 4-1 on page 4-5*, *Table 4-3 on page 4-10* and *Table B-2 on page B-7*.

Average Charge Time

A battery nominally take three hours to charge at ambient temperatures of 30° C or below. Charging at temperatures between 30° C and 35° C results in longer charge times of three to six hours.

NOTE Units do not charge at ambient temperatures above 35° C or below 0° C.

Charging Batteries

Before using the CA50 for the first time, charge the battery using a cradle until the green charge status LED lights. See *Table 4-3 on page 4-10* for charge status indications. For information about the accessories available for the device, see *Appendix B, Accessories*.

NOTE The CA50 does not have an internal battery back up. If the battery is removed or replaced, the device cold boots.

To charge batteries, use one of the following cradles:

- *Single-Slot CA50 Charging Cradle with Spare Battery Charger on page B-5*
- *Eight-Slot Spare Battery Charger on page B-8*.

Charging the Battery in a CA50

To charge the battery in a CA50:

1. Connect the charging accessory to the appropriate power source. See *Appendix B, Accessories* for setup information.
2. Insert the CA50 into a cradle. The CA50 begins charging. The charge status LED is solid red while charging, then turns solid green when fully charged. See *Table 4-3 on page 4-10* for charging indications.

For charging times, see *Average Charge Time on page 3-7*.

Charging Spare Batteries

Use one of the following accessories to charge a spare battery:

- *Single-Slot CA50 Charging Cradle with Spare Battery Charger on page B-5*
- *Eight-Slot Spare Battery Charger on page B-8*.

To charge a spare battery:

1. Connect the spare battery charging accessory to the appropriate power source.
2. Insert the spare battery into the accessory spare battery charging slot with the charging contacts facing down, contacting the charging pins in the cradle.

   The battery begins charging. The charge LED on the cradle lights to show the charge status.

For charging times, see *Average Charge Time on page 3-7*.

Charging Temperature

See *Charging Temperature on page 3-7* for detailed information.

Battery Management Recommendations

Table 3-1 includes recommendations to maximize the CA50 run time per battery charge.

NOTE Ensure batteries are fully charged before use.

Table 3-1 *Battery Management*

CA50 Usage	Battery Management Recommendation
Out-of-Box (no Access Point association)	Associate the CA50 device to an Access Point as quickly as possible. Out-of-box the CA50 lacks network details and is not associated with an Access Point. In this state, the internal radio behaves as though the device is in CAM mode, utilizing maximum radio power until it is associated with an Access Point. The device should not remain in this max power usage mode for an extended period of time to prevent the battery from depleting.
Wireless Profile Creation (associated to Access Point)	Ensure the power mode is set to *Max Power Save*. This parameter is set through the scanning of MSP generated Rapid Deployment bar codes. See pages *12-1* and *F-1*.
Optimizing User Settings	• Use *Ring* mode versus *Vibrate* mode; *Ring* mode consumes less power than *Vibrate* mode. • Disable the vibrate indicator by modifying the user XML profile in the Alert element “RecvCallNoCall” within the Alerts node. The “RecvCallNoCall” element must have a valid token for the value of the VisibleIndicator attribute and an empty string for value of the VibrateIndicator attribute. Example: <Alerts> . . <Alert Event=”RecvCallNoCall” tone=”\Application\Wav\NewMsg.wav” VisibleIndicator=”RedSlowBlink” VibrateIndicator=”” /> . </Alerts> • Use the earbud, when possible, versus the CA50 internal speaker; use of an earbud consumes less power than using the CA50 internal speaker. • Backlight: The backlight is a contributor to battery usage. Excessive backlight usage can contribute to reducing run time. The backlight can be modified using the UCA server Web Console: *System Settings* > *Client.Backlight.Level* (see *System Settings on page 9-37*).
When Not in Use	• Suspend the CA50 by putting it into *sleep mode*. **Note:** When in sleep mode, CA50 power consumption is significantly reduced but not eliminated. In this mode the CA50 can remain out of a charger for approximately 48 hours before requiring a recharge. For more information on charge times see *Charging Batteries on page 3-8*.

Table 3-1 *Battery Management (Continued)*

CA50 Usage	Battery Management Recommendation
Optimizing a Wireless Network for Voice Communication	Using the VoIP capabilities of the CA50 requires the wireless network to be optimized for voice, which places special demands on the network beyond those required for data alone. Many networks in use today are only optimized for data traffic. • Most wi-fi systems are built to deliver a signal strength of -75dBm. This is enough power for laptops to surf the Web and email. • VoIP requires a better signal to maintain a call and should be designed at -65dBm, which assures at least a 20dB SNR, even with a high noise floor. For details on tools to perform a voice centric network site survey, see what is the first step in selling the CA50 Solution?
Maximizing Run Time (wireless infrastructure optimization)	The items below are listed in order of importance with the most important recommendation first: • A high multicast bit rate (set at the Access Point) increases run time while decreasing walkie-talkie range and quality. • Increasing the number of CA50 devices actively engaged in voice (unicast) calls associated with the same Access Point can decrease CA50 run time and voice (unicast) call quality. Loads associated with wireless bar code scanning and walkie-talkie communication are less taxing than voice (unicast) calls. • Increasing the RF activity (from non-CA50 devices, Access Points, noise, etc.) in the area of CA50 usage can decrease the CA50 run time, voice quality and range. • Increasing CA50 distance from an associated Access Point can decrease CA50 run time (due to rate scaling) even when no perceptible degradation in voice quality is noted. ✓**Notes:** [1] When a CA50 is out of range of an Access Point, radio power consumption is maximized, significantly degrading run time/battery life. [2] In 802.11b and 802.11g implementations, it is recommended to use radio channels 1, 6 and 11 to minimize inter-channel interference (which decreases run time).

✓ ***NOTE*** Standard mobile suspend (sleep mode), resume and auto suspend modes used in Motorola mobile devices do not apply to the CA50 voice device because the radio must stay active to transmit walkie-talkie conversations and receive incoming voice calls.

Powering On the CA50 - Pre-deployment

The CA50 device powers on when a charged battery is installed (see *Inserting the Battery on page 3-4*). If a charged battery was already installed and the device is put into sleep mode, press any key to wake the device.

NOTE If the device does not power on, the battery charge may be depleted. See *Charging Batteries on page 3-8*.

Upon initial use, first time out-of-box, the Motorola splash screen displays briefly and then the device boots up to the CA Menu. For additional information about the CA Menu see *CA Menu on page 3-12*.

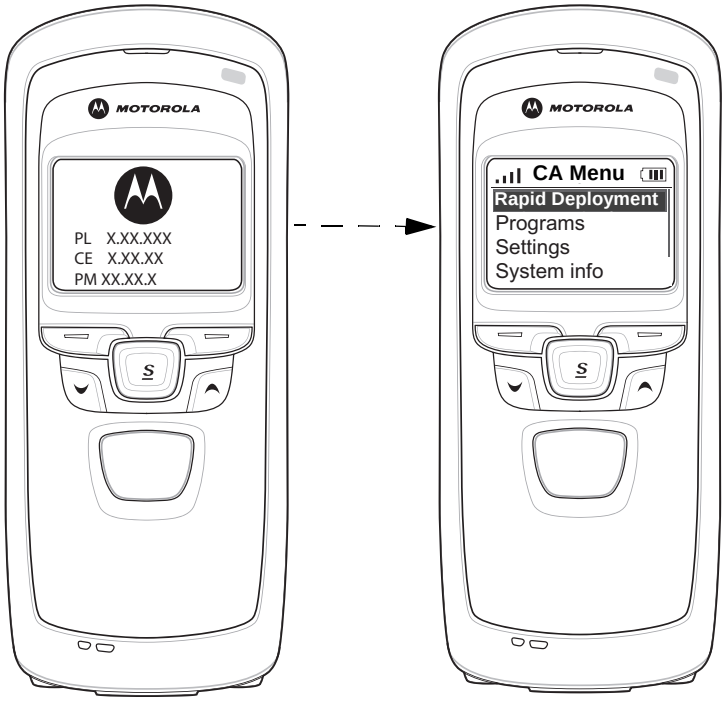


Figure 3-6 *Motorola Splash Screen and CA Menu.*

NOTE Use Rapid Deployment bar codes to initially stage and deploy the CA50 by setting the wireless network configuration and enter the UCA server IP address. For information about configuring the CA50 using Rapid Deployment, see *Chapter 12, Staging and Deploying CA50s* and *Appendix F, Rapid Deployment*.

CA Menu

The CA Menu is the factory programmed start menu that allows access to the Rapid Deployment client, device settings, system information (such as OS version), reboot options and sleep mode (power off). The CA Menu displays when any of the events below are encountered.

- First boot up of the device out-of-box.
- Rebooting the device without UCA server details loaded (for example, the UCA server IP address).
- Rebooting the device and manually pushing override keys that allow access to the CA Menu.

NOTE The CA Menu displays out-of-box and when reconfiguring the device, essentially whenever the CA50 is not running its UCA client application.

From the CA Menu use the **Up** and **Down** arrow keys to highlight a menu option. Press the **Scan/Select (S)** key to launch an option. For a list of CA Menu options, see *Table 4-6 on page 4-19*.

To access the CA Menu, see *Accessing the CA Menu on page 4-18*.

Sleep Mode

NOTE When in sleep mode, CA50 power consumption is significantly reduced but not eliminated. The CA50 can remain out of a charger for approximately 48 hours before requiring a recharge.

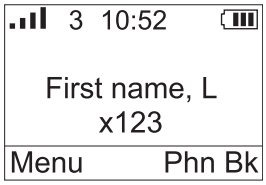
Placing the CA50 into sleep mode prevents the occurrence of communications (e.g., OS upgrades, phone book updates, etc.). If the device is in sleep mode and placed in the cradle, the device wakes up and logs off the user.

To place the CA50 device in sleep mode:

- Select *Sleep* mode from the CA Menu (see *CA Menu Options on page 4-19*).

  or

- On the UCA client application desktop, press the left soft key to display the UCA client application desktop menu (defined by the user profile and the current menu state). Use the **Up** and **Down** keys to select *Sleep* mode.



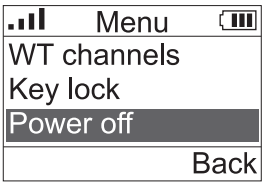


Figure 3-7 *Desktop Display* *UCA Client Application Desktop Menu*

- The *Sleep mode?* confirmation window displays.

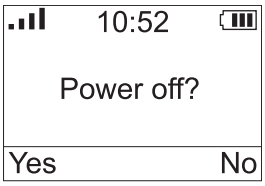


Figure 3-8 *Sleep Mode Confirmation Display*

- Press the left soft key to select *Yes* to turn off the screen and radio and put the device into sleep mode.
- Press the right soft key to select *No* (to avoid sleep mode) and return to the prior screen.

Power Off

Remove the battery to power off the CA50 device.

(Page intentionally blank.)

Chapter 4 CA50 Modes of Operation

Overview

This chapter covers various information about using the CA50, post-deployment.

Valuable topics in this chapter include:

- Logging in and logging out (beginning on *CA50 Login - Post-deployment on page 4-2*)
- Employee identity validation (*page 4-3*)
- Status icons (*page 4-5*)
- Keys, controls and features on the device (*page 4-7*)
- Earbud use (*page 4-13*)
- Scanning (*page 4-15*)
- Resetting (*page 4-17*)
- Desktop menus (*page 4-19*)
- Walkie-talkie (*page 4-31*)
- RF network optimization for walkie-talkie communication (*page 4-31*)
- Naming walkie-talkie channels (*page 4-33*)
- Voice (unicast) calls (*page 4-36*)
- Business applications (*page 4-54*)
- Messaging/paging (*page 11-26* and *page 11-44*).

NOTE 1. For a complete checklist outlining all the development and deployment activities required to implement a CA50 solution, see the *Development and Deployment Checklist on page 2-1*.

2. For a table listing the most commonly requested topics within this manual, see the Quick Startup Guide on the last page of the manual.

CA50 Login - Post-deployment

Login Options

There are two login options for a store location. Only one method can be used at a single location.

- Scan bar codes login:CA50 users scan a bar code to log into the device. User accounts are created on the server application. The login bar code sheet must be generated by a third party application.
- Numeric input login: CA50 users enter a unique four digit number that is tied to a specific user account. Input is supported through a ‘spin box’ UI.

NOTE The numeric input can be a number from 1 to 5 digits.

If a device must be redeployed, an administrator must access the CA Menu and launch Rapid Deployment, or perform another deployment option.

See *Powering On the CA50 - Pre-deployment on page 3-11* for information about pre-deployment.

See *Appendix F, Rapid Deployment* for information about loading software onto the CA50 and configuring the device for use.

Power On

1. The CA50 device powers on when a charged battery is installed (see *Inserting the Battery on page 3-4*). If a charged battery was already installed and the device is put into sleep mode, press any key to wake the device.

NOTE If the device does not power on, the battery charge may be low or depleted. See *Charging Batteries on page 3-8*.

2. The Motorola splash screen displays during boot. The device ID and boot type also display.

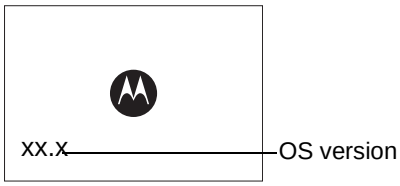


Figure 4-1 *Motorola Splash Screen*

NOTE Because login options are server configurable (see *Login Options on page 4-2*) an automatic login can be enabled on the server and the screen samples that follow may not display at login.

1. The device attempts to connect to the network, negotiating the network connection. On this screen, the user can press the right soft key to *Cancel* connection negotiation and put the device into sleep mode.

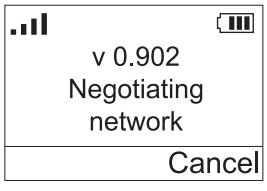


Figure 4-2 *Network Negotiation Screen*

2. The device connects to the UCA server. On this screen, press the right soft key to *Cancel* the connection and to put the device into sleep mode.

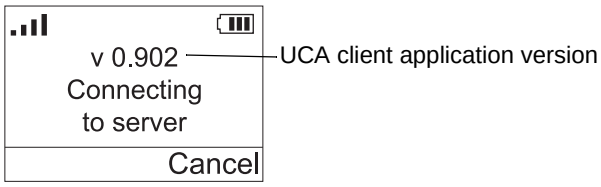


Figure 4-3 *Server Connection Screen*

3. Upon successful login, the UCA client application can display a login prompt. The login prompt is server configurable (see *Login Options on page 4-22*).

NOTE If the device cannot connect to the network or find the UCA server during the login attempt, an error message displays, a beep sounds and a red LED blinks. A countdown error screen appears. When the countdown is complete, an automatic retry is attempted. If the **Log** key is pressed from the countdown screen, a *Log* screen displays with a log of the attempted network connection error. Automatic retry is disabled while in the *Log* screen.

From the *Log* screen a user can press the right soft key to go *Back*: to the error screen and restart automatic retry, or press right soft key to *Cancel* and put the device into sleep mode.

Employee Identity Validation (Scanning a Login Bar Code and entering a PIN)

For an employee to log in to the UCA server, scan an employee login bar code and optionally enter a PIN. For additional details see *page D-15* and *E-14*.

NOTE The Employee Identity Validation process is controlled by the UCA server application. For details see *page 5-1*.

1. When the device is logged off, a log in prompt screen displays.

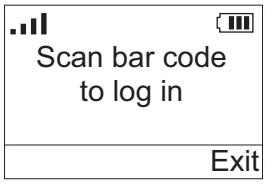


Figure 4-4 *Login Prompt*

2. Press the **Scan/Select** key to scan the appropriate login bar code. Upon a successful login, a decode beep sounds and the green LED blinks.

3. If a PIN is required to continue, the PIN input screen displays.

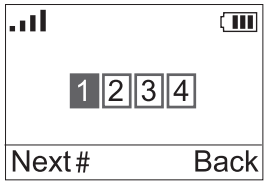


Figure 4-5 *PIN Input*

NOTE PIN entry can be used to enhance security and assist in employee identity validation. For a sample implementation see *page E-14*.

4. Press the **Up** and **Down** keys to increment numbers in the current number field.
5. Press the left soft key to advance to the *Next* number field and press the **Up** and **Down** keys to increment numbers in the current number field.
6. Repeat until the correct PIN is entered in each number field.
7. Upon successful login, the desktop displays.

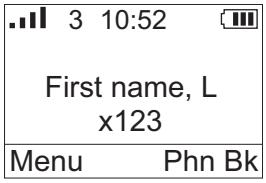


Figure 4-6 *Successful Login - Desktop*

Logoff Options

There are several log off options:

- Scan logoff bar code: Scan a specified bar code to log off. This logoff method only works when the device is at the desktop screen.
- Menu selection logoff: Select the *log off* option from the menu.
- Cradle logoff: Put the device into the charge cradle to logoff. Logoff is triggered by detecting an AC power connection.

Upon initial use, initial boot or after a reboot, the Motorola splash screen displays for approximately 10 seconds and the device boots up to a text-based interface to CA Menu that displays the *Start* menu.

Display Status Icons

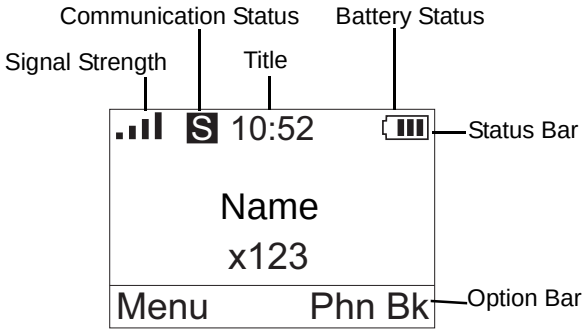


Figure 4-7 *CA50 Display Screen*

Table 4-1 *CA50 Display Icons CA50/UCA Client & Server Developer & User Guide*

Icon	Description
Wireless Signal Strength	
	Excellent wireless network signal (100%-75% signal).
	Fair wireless network signal (50%-25% signal).
	No wireless network signal.
Communication Status	
1 - 32	Walkie-talkie communication transmit channel number. A number from 1-32 channels displays in walkie-talkie communication non-scan mode.
	Channel number. Single channel walkie-talkie communication mode.
	Walkie-talkie communication scan mode.
	On Call icon. Displays when on an active voice (unicast) call.
	Mute mode. Displays when speaker is muted and when microphone is muted while on a voice (unicast) call.
	Do not disturb.
Title	
10:52	Time of day in hh:mm format (default); 12 hour clock. On a menu, phone book or business application screen a title displays. The title is one to four characters in length. If no title is available, the default is time of day.

Table 4-1 *CA50 Display Icons CA50/UCA Client & Server Developer & User Guide (Continued)*

Icon	Description
Battery Strength	
	Battery Strengths (from left): • Battery is fully charged (100% - 75% charge remains) • 75% - 50% charge remains • 50% - 25% charge remains • 25% - 10% charge remains • 10% - no charge remains • When the capacity of the battery reaches this level the battery icon blinks and an error tone sounds once every minute.
	AC Power - Battery is charging
	Battery capacity is unknown
	On USB - Battery Installed On USB - No Battery Installed AC Power - No Battery Installed AC Power - Charge Complete
	AC Power - Too Cold to Charge AC Power - Too Hot to Charge AC Power - Charger Fault

CA50 Keys, Actions and Features

Table 4-2 *CA50 Key Functionality, Device Actions and Features*

Key/Action/Feature	Description
KEYS	
Soft Keys	Left and right soft key functions are identified by the screen display.
Scan/Select (S) key	A **Select** key when a menu is activated; a **Scan** key to scan bar codes when not in a menu. This key can be programmed to scan the bar code and post the data to the server when the device is in desktop mode. The URL to post the data can be modified/updated in the *System Settings* tab of the UCA server console. The developer can only reprogram this key for business applications when the device is in the desktop mode. The **Scan** key launches the scan engine and posts decoded data to a server-defined URL. By default, it posts to the sample item lookup application. This default applies system-wide with the possible exception of within a business application where developers can override the default. Press the **Mute** and **Scan/Select (S)** keys simultaneously to perform warm and cold boots.
Front Application Key (Walkie-Talkie)	**Walkie-Talkie** key by default. The developer cannot override the functionality in a business application. The key can be defined as a hotlink from the UCA Client. See *Chapter 7, XML Profile for Configuration*.
Side Application Key (Walkie-Talkie)	**Walkie-Talkie** key by default. The developer cannot override the functionality in a business application. The key can be defined as a hotlink from the UCA Client. See *Chapter 7, XML Profile for Configuration*.
Up and Down Arrow Keys	Always scroll, but are server configurable for the desktop only. When in desktop mode, these keys can be configured to launch a business application.
Volume Keys	Up: Increases speaker or receiver volume, depending on the state of the device. Down: Decreases speaker or receiver volume, depending on the state of the device.
Mute Key	Mutes speaker or microphone, depending on the state of the device. Press the **Mute** and **Scan/Select (S)** keys simultaneously to perform warm and cold boots.

Table 4-2 *CA50 Key Functionality, Device Actions and Features (Continued)*

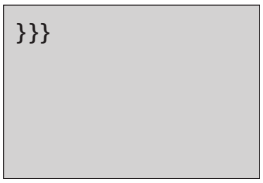
Key/Action/Feature	Description
ACTIONS	
Boot - Warm	Use the **Up** and **Down** keys and select *Warm Boot* from the CA Menu. or Press and hold the **Mute** and **Scan/Select (S)** keys simultaneously (for button locations, see *page 3-2*) until a blank screen displays. Release the buttons while the red LEDs are flashing. This takes approximately 3 seconds. See *Performing a Warm Boot on page 4-17* for detailed information.
Boot - Cold	Remove and re-insert the battery. or Use the **Up** and **Down** keys to select *Cold Boot* from the CA Menu. or Press and hold the **Mute** and **Scan/Select (S)** keys simultaneously until the screen displays the characters **}}}** (shown below). When these characters display release the **Mute** and **Scan/Select (S)** keys. Failure to release these keys re-initiates a cold boot.  See *Performing a Cold Boot on page 4-17* for detailed information.
Calls - Walkie-Talkie	Press either the side or front **Walkie-Talkie** key.
Calls - Voice	Find a name in the phone book on the device and press the **Scan/Select (S)** key. Note: The phone book is configurable via the server application.
Device on	Press any key to turn the device on.
Device off	Select *Sleep mode* from the CA Menu. Note: Menu items are configurable via the server application. The sleep mode feature may not apply to all environments. **Note:** When in *sleep mode*, CA50 power consumption is significantly reduced and can remain out of a charger for approximately 48 hours before requiring a recharge. Placing the CA50 into sleep mode prevents the occurrence of communications (e.g., OS upgrades, phone book updates, etc.). If the device is in sleep mode and placed in the cradle, the device wakes up and logs off the user.
Mute microphone	Press the **Mute** key. Note: The microphone can only be muted while on a voice call.
Unmute microphone	Press the **Mute** key. Note: The microphone can only be muted while on a voice call.

Table 4-2 *CA50 Key Functionality, Device Actions and Features (Continued)*

Key/Action/Feature	Description
Mute speaker	Press the **Mute** key. Note: The speaker can only be muted when the user is not on a voice call.
Unmute speaker (if muted)	Press the **Mute** key.
Scan a bar code	Press the **Scan/Select (S)** key. Note: Usually used to scan a bar code except when in a menu list or on a tumbler or a scrollable information screen.
Volume - Increase receiver	Press the **Volume Up** key. Note: Receiver volume can be adjusted while on a voice call only.
Volume - Decrease receiver	Press the **Volume Down** key. Note: Receiver volume can be adjusted while on a voice call only.
Volume - Increase speaker	Press the **Volume Up** key. Note: Speaker volume can be adjusted from any screen, except when on a voice call.
Volume - Decrease speaker	Press the **Volume Down** key. Note: Speaker volume can be adjusted from any screen, except when on a voice call.
CA Menu access	Press and hold the left soft key at boot time.
FEATURES	
Connectors	
ActiveSync	USB Mini-B
Charge Contacts (2)	Charging contacts for cradle connections.
Earbud	Earbud (audio) jack (2.5mm).
Communication	
Receiver	Sound out for voice calls.
LED	Green, red and amber.
Microphone	Sound in for walkie-talkie and voice (unicast) calls.
Speaker	Sound out for walkie-talkie and notifications
Scan Window	Emits laser for scanning bar codes.

CA50 Indications

LED Indications

Table 4-3 *CA50 LED Indications*

Event	LED Status
Walkie-Talkie Indications	
Start walkie-talkie communication transmit	Off
Start walkie-talkie communication receive	Off
Voice (Unicast) Call Indications	
Receive a call	Blink green during ring
Make a call	Off
Busy	Off
Scanning Indications	
Active scanning/laser on	Off
Successful decode	Single green blink
Network Indications	
Connecting to network/server	Off
Connected to UCA server	Single green blink
Connection failure	Slow red blink
No connection/lost connection	Slow red blink
Battery Charging Indications **Note:** CA50 LED (lights only when the device is in cradle.	
Charge in progress	Solid red
Charge complete	Solid green (When the AC power cord is removed, LED turns off.)
Charge error	Red blink
No AC power	Off
LED Indications	
Notification Indications	The device LED indications on the notification message are configurable through profile as well as from the messaging API. See *SendNotification on page 11-26*.
Note: All LED indications are on the CA50 device unless otherwise noted.	

Audio Indications

Table 4-4 *CA50 Audio Indications*

Event	Audio Status
Walkie-Talkie Indications	
Start Walkie-Talkie communication transmit	Start walkie-talkie communication transmit tone
Start Walkie-Talkie communication receive	Start walkie-talkie communication transmit tone
Voice (Unicast) Call Indications	
Receive a call	Ring tone (end-user configurable)
Make a call	Ringing tone (via receiver)
Busy	Busy tone (via receiver)
Scanning Indications	
Active scanning/laser on	No sound
Successful decode	Successful decode tone
Network Indications	
Connecting to network/server	No sound
Connected to UCA server	Connection tone
Connection failure	Single ‘error’ tone
No connection/lost connection	Single ‘error’ tone
Battery low indicator	Plays an audio file (configurable in the profile)
Power Indications	
Low battery	Beeps every 60 seconds when the battery is 10 percent (or lower)*
Screen Indications	
Low battery	Blinking battery icon
Audio Indications	
Notification Indications	Audio indications for notification messages on the device are configurable through the profile as well from the messaging API. See *SendNotification on page 11-26*.

*The enable/disable of the low battery beep and the beep interval are controlled by registry keys:

[HKEY_LOCAL_MACHINE\Software\Motorola\UCAShell\UCAClient]

BatteryIconBeepEnable"=dword:1; 0 - Do not beep, 1 beep on low batt

BatteryBeepInterval"=dword:3C; time in seconds for the low battery sound to beep

Screen Indications

Table 4-5 *CA50 Display Screen Indications*

Event	Display Status
Walkie-Talkie Indications	
Start walkie-talkie communication transmit	No display
Start walkie-talkie communication receive	No display
Voice (Unicast) Call Indications	
Receive a call	Caller ID
Make a call	Outbound caller ID
Busy	Outbound caller ID
Low battery	The low battery image blinks
Scanning Indications	
Active scanning/laser on	No display
Successful decode	Application dependent; no display by default
Network Indications	
Connecting to network/server	Connecting message
Connected to UCA server	UCA client application
Connection failure	Error message
No connection/lost connection	Error message
Lost Connection	Device can be configured to Beep or log the loss of connection, disabled by default*
*Option to Beep or Log the loss of RF Signal. This feature is controlled by the following reg keys: [HKEY_LOCAL_MACHINE\Software\Motorola\UCAShell\UCAClient] "SignalLossBeep"=dword:0 ; 0 - No Beep, 1 - Beep for signal loss "SignalLossLog"=dword:0 ; 0 - No Log, 1 - Log of signal loss	
Battery Charging Indications	
Charge in progress	Battery icon in charging state
Charge complete	Battery icon fully charged
Charge error	Charge error message

Table 4-5 *CA50 Display Screen Indications (Continued)*

Event	Display Status
No AC power	Battery icon
Other	
Do not disturb	Status bar icon and desktop text

Using an Earbud

You can use a mono earbud for audio communication when using an audio-enabled application. To use an earbud, plug the earbud (2.5mm jack) into the mono audio connector on the side of the CA50. Plugging an earbud into the jack mutes the speaker.

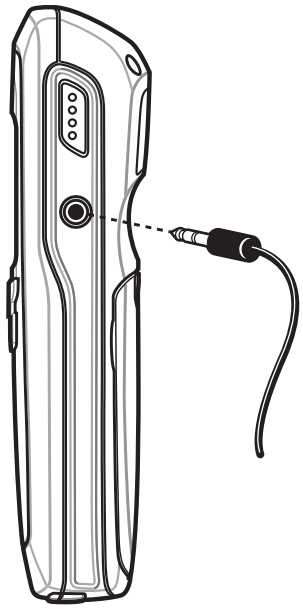


Figure 4-8 *Mono Earbud Connection*

Wireless Security Protocols for the CA50

CA50 wireless security is based on underlying Fusion support. At each CA50 release, a subset of voice-friendly security modes are validated for use with CA50 voice.

Currently, the CA50 supports the following security protocols. The most secure are listed first:

- WPA2-AES (see note 1)
- WPA-TKIP (see note 1)
- WEP 128 (see note 3)
- WEP 40
- Open (see note 2)

WPA2-AES is the most secure; in addition, it is typically slightly faster than WPA-TKIP due to improvements in the protocol.

WPA-TKIP provides a good balance between security and roam times. It can also serve as stand-in for WPA2-AES for advanced debugging: WPA-TKIP can be decrypted with a full wireless snooper capture, while WPA2-AES cannot.

NOTE 1 The use of an authentication server is not recommended for voice.
NOTE 2 Recommended for diagnosis and fast roaming.
NOTE 3 Consider if it meets security needs. WEP is a natural fast roamer.

Data Capture

The CA50 offers laser scanning for 1D bar codes.

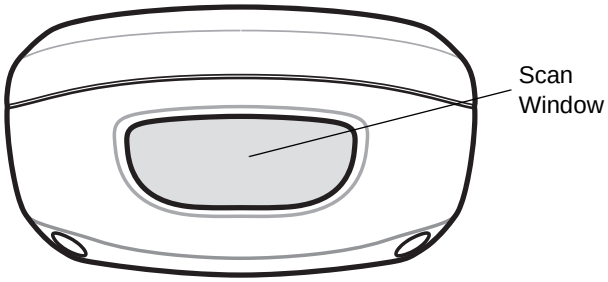


Figure 4-9 *Scan Window*

Laser Scanning

The CA50 has an integrated laser scanner with the following features:

- Reads a variety of bar code symbologies, including the most popular linear, postal, and 1D code types.
- Intuitive aiming for easy point-and-shoot operation.

Scanning Considerations

Typically, scanning is a simple matter of aim, scan, and decode and a few quick trial efforts master it. However, consider the following to optimize scanning performance:

- Range

  Any scanning device decodes well over a particular working range — minimum and maximum distances from the bar code. This range varies according to bar code density and scanning device optics.

  Scanning within range brings quick and constant decodes; scanning too close or too far away prevents decodes. Move the scanner closer and further away to find the right working range for the bar codes being scanned.

- Angle

  Scanning angle is important for promoting quick decodes. When laser beams reflect directly back into the scanner from the bar code, this specular reflection can “blind” the scanner.

  To avoid this, scan the bar code so that the beam does not bounce directly back. But don’t scan at too sharp an angle; the scanner needs to collect scattered reflections from the scan to make a successful decode. Practice quickly shows what tolerances to work within.

- Hold the CA50 farther away for larger symbols.
- Move the CA50 closer for symbols with bars that are close together.

NOTE Scanning procedures depend on the application and CA50 configuration. An application may use different scanning procedures from the one listed above.

Scanning

1. Ensure that a scan enabled application is loaded on the CA50.
2. Aim the scan window at the bar code.

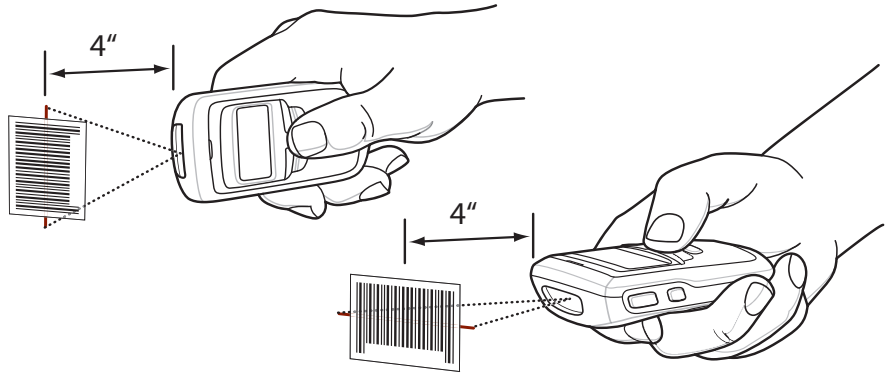


Figure 4-10 *Laser Scanning*

3. Press the **Scan** key. Ensure the red scan beam covers the entire bar code. A single green LED blink, by default, to indicate the bar code was decoded successfully.

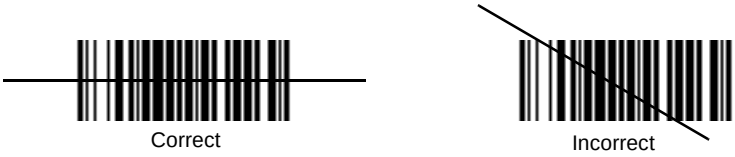


Figure 4-11 *Laser Scanner Aiming Pattern*

Resetting the CA50

There are two reset functions, warm boot and cold boot. Perform a warm boot first. If the device still does not respond, perform a cold boot.

Neither a warm nor cold boot clears deployment settings such as network login and UCA server IP address.

Performing a Warm Boot

The state of the device (pre-deployment or post-deployment) determines the display when the warm boot completes. If the state of the device is pre-deployment, by default the device boots into the CA Menu.

If the state of the device is post-deployment, by default the device boots into the UCA client application and attempts to connect to the UCA server. To boot into the CA Menu when the boot completes, the user must press and hold the left soft key during the boot.

There are two ways to perform a warm boot:

- Use the **Up** and **Down** keys and select *Warm Boot* from the CA Menu.

  or

- Press and hold the **Mute** and **Scan/Select (S)** keys simultaneously until a blank screen displays and the red LED flashes. This takes approximately 3 seconds.

IMPORTANT When the blank screen displays the user must release the keys when the warm boot initiates to avoid initiating a cold boot.

Performing a Cold Boot

The state of the device (pre-deployment or post-deployment) determines what is displayed when the cold boot completes.

- If the state of the device is pre-deployment, by default the device boots into the CA Menu.If the state of the device is post-deployment, by default the device boots into the UCA client application and attempts to connect to the UCA server. To boot into the CA Menu when the boot completes, the user must press and hold the left soft key during the boot.

There are three ways to perform a cold boot:

Use the **Up** and **Down** keys and select *Cold Boot* from the CA Menu.or

- Press and hold the **Mute** and **Scan/Select (S)** keys simultaneously until the screen displays the characters **}}}** (shown in *Figure 4-12*). When these characters display release the **Mute** and **Scan/Select (S)** keys. Failure to release these keys re-initiates a cold boot.

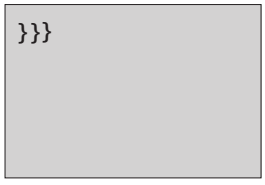


Figure 4-12 *Cold Boot Display*

or

- Remove and reinsert the battery.

Waking the CA50

When the device is in sleep mode (suspended), wake the device by pressing any key.

Accessing the CA Menu

To access the CA Menu when an application and/or network settings are already loaded on a CA50, for example after the CA50 is deployed in a store environment:

1. Warm boot the device: Press and hold the **Mute** and **Scan/Select (S)** keys simultaneously (for button locations, see *page 3-2*) until a blank screen displays. Release the buttons while the red LEDs are flashing. This takes approximately 3 seconds.
2. **IMMEDIATELY WHILE THE RED LED IS STILL FLASHING**, press and hold the left soft key until the CA Menu screen displays. **DO NOT RELEASE** the left soft key until the CA Menu displays and remains on the display for 5 seconds.

For details about the CA Menu, see *CA Menu on page 3-12*.

Out-of-Box CA Menu Options

Table 4-6 lists the out-of-box menus available on the CA Menu when the device is powered on. Use the left and right soft keys on the device to highlight menu options and press the **Scan/Select (S)** key to launch the highlighted option.

Table 4-6 *CA Menu Options*

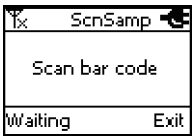
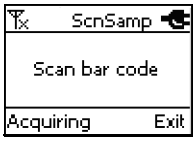
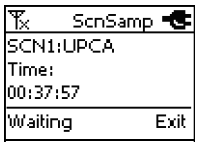
Menu	Description
Rapid Deployment	Prior to use, the CA50 device must be configured. The Rapid Deployment client facilitates software downloads to the CA50 client. For detailed information about the Rapid Deployment Client, see *Appendix F, Rapid Deployment*.
Programs Note: There are several out-of-box programs on the CA50 when the device is powered on. The programs listed below are highlights.	
UCA Client Application	Invokes the UCA client application which attempts to connect to the UCA server.
ScanSamp	ScanSamp (scan sample application) prompts the user to scan a bar code and the scanned data displays. Scroll up and down to display all available data. Press **Scan/Select** again to scan another bar code, or press the right soft key to exit and return to the previous window. Example of scanned data displayed: Data: 65830923843078; Type: 0x32; Source: SCN:UPCA; Time: 00:00:22. This application can also be used to test the laser if it does not emit when the **Scan/Select** is pressed. To scan a bar code: **1.** On the *CA Menu*, select *Programs* > *ScanSamp*. **2.** Press **Scan/Select** to launch the ScanSamp test application.  **3.** Aim the scan window at a bar code and press **Scan/Select**. The laser emits, the lower left of screen displays *Acquiring*, and the red LED remains on until the data is acquired or the laser times out.  **4.** View scan data on the display. On a good decode, the green LED lights for approximately 3 seconds and the scan data displays. Use the down arrow to scroll through the information.  See *Troubleshooting the CA50 Device on page 14-11* for information about scanning and laser problems and solutions.

Table 4-6 *CA Menu Options (Continued)*

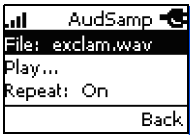
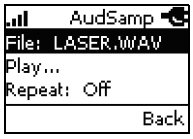
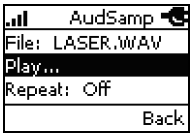
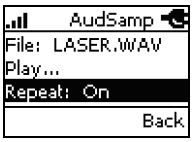
Menu	Description
Audio Test App	Audio Test allows the user to play and record sounds and select the output type (speaker or handset). This application can also be used to test the audio play logic for the speaker and the receiver and can also test the audio record using the microphone. Select *Play Audio* to choose a sound file to play from various sources (e.g., network, Platform folder, Application folder, etc.). To play and record sounds: **1.** On the *CA Menu*, scroll to select *Programs* > *Audio Test App*. The application executes and displays the *AudSamp* screen. **2.** Scroll to *Output: Handset* and press **Scan/Select**. The output option toggles from to *Handset* to *Speaker.* **3.** Scroll up to *Play Audio* and press **Scan/Select** to display the *AudSamp Play... screen*.  **4.** Press **Scan/Select** to select the *File:* option. The *Select File* screen displays. **5.** On the *Select File* screen navigate to a file to play and press **Scan/Select** to select the highlighted file. The *AudSamp Play... screen* displays the selected file.  **6.** Scroll down to *Play...* and press **Scan/Select** to play the .wav file. When the *Repeat* option is set to *Off*, the selected audio file plays one time.  **7.** If the *Repeat* option is set to *Off*, scroll down to *Repeat:* and press **Scan/Select** to toggle *Repeat* ON and OFF. Set *Repeat* to *On*  (Audio Test App continues on the next page)

Table 4-6 *CA Menu Options (Continued)*

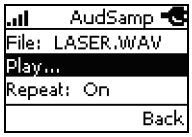
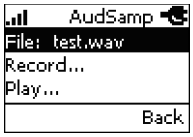
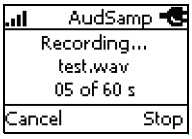
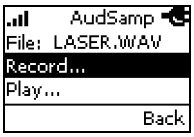
Menu	Description
Audio Test App (continued)	**8.** Scroll up to *Play...* and press **Scan/Select** to play the .wav file. When the *Repeat* option is set to *On*, the selected audio file plays continually.  **9.** Press the left soft key to stop the audio file from playing. **10.** Press the right soft key to go back to the main *AudSamp* screen. **11.** On the main *AudSamp* screen (above) scroll down to *Record Audio* and press **Scan/Select** to display the *AudSamp Record* screen (below).  **12.** Scroll down to *Record...* and press **Scan/Select**. Recording begins. Play a sound into the CA50 receiver. The recording sound saves to the test.wav file.  **13.** At any time, press the right soft key to stop recording.  **14.** When recording stops scroll down to *Play...* and press **Scan/Select** to play the recording. See *Troubleshooting the CA50 Device on page 14-11* for information about audio problems and solutions.

Table 4-6 *CA Menu Options (Continued)*

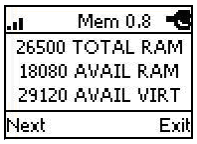
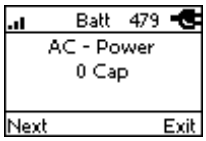
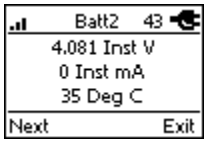
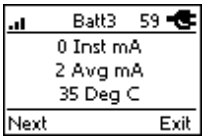
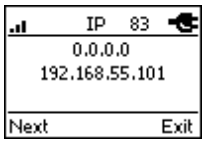
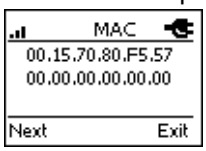
Menu	Description
UCA Sys Info	The UCA Sys Info application can be used to test a variety of situations. • *Mem* screen: If the system is sluggish, the memory screen displays whether or not resources are low.  • *Batt* screens: Three battery screens display battery status including power source, temperature and instantaneous voltage and current draws. If charging issues develop, these screens display an *over temperature* condition.    • *IP* screen: Displays the IP of the network adaptors including USB if the cable is attached. The USB IP is 192.168.55.101. If the WLAN device has an IP of 0.0.0.0, it is not connected to an AP. If there is no IP address at all for the WLAN adaptor, there is a problem communicating with the radio.  • *MAC* screen: Displays the MAC address of the network adaptors. The USB MAC is all zeros. The WLAN adaptor MAC is in the format 00:15:70:XX:XX:XX. If there is no MAC for the WLAN adaptor, there is a problem communicating with the radio.  • ⚠ *IMPORTANT* **Note:** The CA50 must be used for **RF** site surveys that require optimization of CA50 voice communications. For these sites the CA50 is the only acceptable survey tool. Other mobile devices (such as an MC50) do not produce an equivalent signal strength indication as displayed on the CA50. • ***wlan* screen RF Site Survey Tool**: Displays the active profile and monitors **RF** signal strength. The signal strength, displaying the RSSI value in dB, is updated every few seconds. A user can walk from area to area with the device and the display updates show the coverage in different areas.

Table 4-6 *CA Menu Options (Continued)*

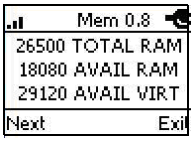
Menu	Description
UCA Sys Info (continued)	To access UCA Sys Info screens: **1.** On the *CA Menu*, scroll to select *Programs* > *UCA Sys Info*. The application executes and displays the *Mem* screen.  **2.** Press the left soft key to display the next screen (*Batt* screen). **3.** Continue pressing left soft key to scroll through all the UCA Sys Info screens. **4.** Press the right soft key to exit UCA Sys Info.

Table 4-6 *CA Menu Options (Continued)*

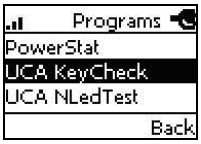
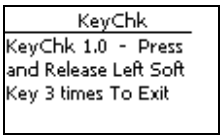
Menu	Description
UCA KeyCheck	This application can be used to test the keys on the device. To ensure all CA50 keys are functioning properly: **1.** On the *CA Menu*, scroll to select *Programs* > *UCA KeyCheck*.  **2.** The application executes and the *KeyChk* screen displays with the version and instructions for exiting the program.  **3.** To verify each key on the device is operating correctly, press the keys listed below to display the appropriate key data. ***NOTE*** For each key press, three sets of information are displayed and the corresponding key code is in the last two lines. Example: For **Walkie-Talkie** key, the correct key code is 193, as shown below: • Press and release the **Down** arrow key - Key Down 40 and Key Up 40 displays. • Press and release the **Up** arrow key - Key Down 38 and Key Up 38 displays. • Press and release **Scan/Select** key - Key Down 13, WM_CHAR 13 and Key Up 13 displays. • Press and release left soft key - Key Down 37 and Key Up 37 displays. • Press and release right soft key - Key Down 39 and Key Up 39 displays. • Press and release the **Volume Up** key - Key Down 175 and Key Up 175 displays. • Press and release the **Volume Down** key - Key Down 174 and Key Up 174 displays. • Press and release the **Mute** key - Key Down 173 and Key Up 173 displays. • Press and release the side **Application** key - Key Down 194 and Key Up 194 displays. • Press and release the front **Walkie-Talkie** key - Key Down 193 and Key Up 193 displays. **4.** To exit the application, press and release the left soft key three times.

Table 4-6 *CA Menu Options (Continued)*

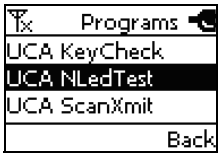
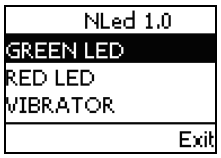
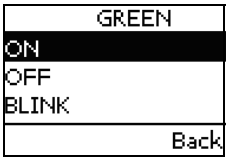
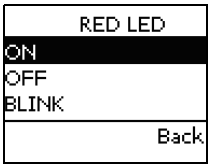
Menu	Description
UCA NLedTest	This application can be used to test the green and red LEDs and the vibrator motor on the device. To test the LEDs and vibrator motor on the CA50 device: **1.** On the *CA Menu*, scroll to select *Programs* > *UCA NLedTest*.  **2.** The application executes and the main *NLed* screen displays.  **3.** On the main *NLed* screen scroll to *GREEN LED* and press **Scan/Select** to display the *GREEN* screen.  **4.** Scroll to *ON* and press **Scan/Select**. The green LED turns on. **5.** Scroll to *OFF* and press **Scan/Select**. The green LED turns off. **6.** Scroll to *BLINK* and press **Scan/Select**. The green LED blinks on and off five times then turns off. **7.** Press the right soft key to go back to the main *NLed* screen. **8.** On the main *NLed* screen scroll to *RED LED* and press **Scan/Select** to display the *RED* screen.  **9.** Scroll to *ON* and press **Scan/Select**. The red LED turns on. **10.** Scroll to *OFF* and press **Scan/Select**. The red LED turns off. **11.** Scroll to *BLINK* and press **Scan/Select**. The red LED blinks on and off five times then turns off. **12.** Press the right soft key to go back to the main *NLed* screen. (UCA NLedTest continues on the next page)

Table 4-6 *CA Menu Options (Continued)*

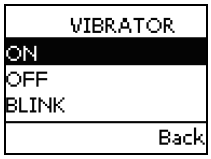
Menu	Description
UCA NLedTest (continued)	**13.** On the main *NLed* screen scroll to *VIBRATOR* and press **Scan/Select** to display the *VIBRATOR* screen.  **14.** Scroll to *ON* and press **Scan/Select**. The vibrator motor turns on. **15.** Scroll to *OFF* and press **Scan/Select**. The vibrator motor turns off. **16.** Scroll to *BLINK* and press **Scan/Select**. The vibrator motor turns on and off five times then turns off. **17.** Press the right soft key to go back to the main *NLed* screen.

Table 4-6 *CA Menu Options (Continued)*

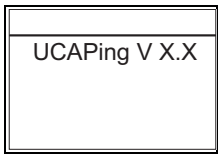
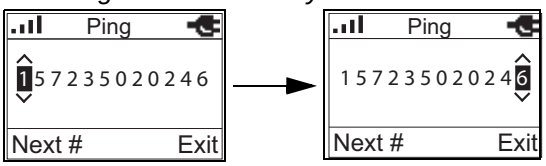
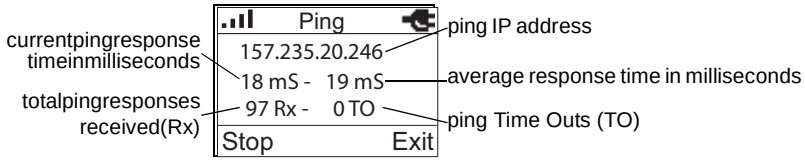
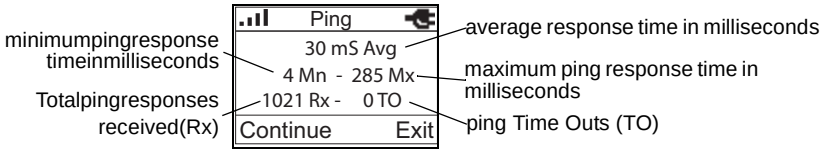
Menu	Description
UCAPing	This application can be used to verify that the device can reach another device and record the response time. To ping another device from your CA50 device: • On the *CA Menu*, scroll to select *Programs* > *UCAPing*. The application executes and the *UCAPing Version* screen displays for 2 seconds.  **1.** The *Ping IP Address Entry* screen follows the *Version* screen.  **a.** Enter the IP address of the device to ping by using the **Up** and **Down** arrows to modify a number and the left soft key to move the cursor to the next number to modify. Continue until the appropriate IP address displays. **Note:** You must enter all 12 digits for an IP address. For example, the IP address 10.1.14.128 should be entered as 010001014128. **2.** When the IP address is complete press the **Scan/Select** key to start *pinging*. **3.** During the Ping operation, the following screen displays.  **4.** Press the left soft key (*Stop*) to suspend current Ping activity. On the *Stop* status screen press the left soft key again to reset the counters and continue pinging.  **5.** Press the right soft key to exit the Ping application.
Airbeam Client	For detailed information about the AirBEAM Client, refer to Airbeam documentation.
Settings	
Backlight Settings	Press **Scan/Select** to modify the setting below; press **Up/Down** keys to scroll options list. • Disable/Enable Backlight • Brightness Level (of the display): Dimmest; Dim; Bright; Brighter; Brightest • Backlight Timeout: 2 sec; 5 sec; 10 sec; Always On • On with Keypress: Yes; No

Table 4-6 *CA Menu Options (Continued)*

Menu	Description
Scanner	Press **Scan/Select** to modify the setting below; press **Up/Down** keys to scroll options list. • Scnr Name: Displays the name given to the device. • Reader Params: Displays Reader Type; Aim Type; Aim Duration; Aim Mode; Beam Width; Beam Timer; Redun; Security Level. Note: Not all Reader Parameter settings can be modified. • Interface Params: Displays Interface Type; Enbl Stl Time; Pwr Stl Time. • Scan Params: Displays ID Type; Scan Type; Local Feed; Dcd Beep Time; Dcd Beep Freq; Dcd LED Time; Dcd WAV File; Start Beep Time; Start Beep Freq; Start LED Time; Start WAV File; Inter Beep Time; Inter Beep Freq; Inter LED Time; Inter WAV File; Fatl Beep Time; Fatl Beep Freq; Fatl LED Time; Fatl WAV File; NFatl Beep Time; NFatl Beep Freq; NFatl LED Time; NFatl WAV File; Act Beep Time; Act Beep Freq; Act LED Time; Act WAV File.

Table 4-6 *CA Menu Options (Continued)*

Menu	Description

Network	Press Scan/Select to modify the setting below; press Up/Down to scroll options list.
	• Network Status - Displays Active Profile and IP Address
	• Manage Profiles - Add or manage existing profiles
	  • Add - To add a profile, you must create test bar codes using a third party tool for Profile Name and SSID, as well as other fields that require data input.
	  • Options for existing profiles:
	  • Disable - Disable profile. Profile remains in the list.
	  • Edit Profile - Only profiles entered with the **Add Profile** option can be edited. Profiles scanned in with RD bar codes or imported are encrypted and the details cannot be viewed.
	  • Connect - Connect to this profile.
	  • Move Up - Move up in the profile list.
	  • Move Down - Move down in the profile list.
	  • Delete - Delete profile from registry.
	  Note: If profile was scanned in or exported, a reg key is created on the CA50. If this reg key is not deleted, the profile remains on the device and appears in the list after a cold boot. Rapid Deployment creates the following two reg keys on the CA50 in the \Application directory:
	  FuPBAPIPluginCBO.reg
	  FuPBAPIPluginCBP.reg
	• Export Profiles - Export current profiles into reg key(s).
	• Import Profiles - Import profile reg key(s).
	• Adaptor Params - Adaptor Type and MAC address of radio.
	• Select 802.11 Band - Enable/Disable B/G and/or A band (B/G only is on by default).
	• Select 802.11d Options - Enable/Disable 802.11d (is on by default).
	• Enable Profile Roaming - Profile Roaming is disabled by default.
	• Fusion Version Info - Version numbers of the Fusion Components.
	• Save Fusion Log - This saves a snapshot of the Fusion log to a file in the \Application folder called FusionLog.txt.

Certificates	Manage certificates.
Date and Time	Set date, time, and time zone on the device.
System Properties	View and change program store and object store memory allocations.
System Info	
UUID	Universal Unique Identifier of CA50 device
OS Version xx.xx.xxxx	CE core version number.
OEM Name xxx	Motorola.
OEM Version x.xx.xx	Build revision for CA50 software.

Table 4-6 *CA Menu Options (Continued)*

Menu	Description
BL xx.xx.xxxx	Monitor code version.
PM xx.xx.xxxx	Power micro version.
Display	Display specifications such as type, resolution, colors, BBP, and adaptor version information.
Battery Info	Display power source, battery type, and battery charge.
Misc	Hardware information such as clock speeds and flash ID.
DLL/EXE Versions	Version information for all drivers and executables on the device that respond to the *GetVersionInfo* API.
Sleep Mode	Suspend the device (see *Sleep Mode on page 3-13*). ✓ **Note:** When in *sleep mode*, CA50 power consumption is significantly reduced and can remain out of a charger for approximately 48 hours before requiring a recharge. Placing the CA50 into sleep mode prevents the occurrence of communications (e.g., OS upgrades, phone book updates, etc.). If the device is in sleep mode and placed in the cradle, the device wakes up and logs off the user.
Cold Boot	Cold boot the device (see *Resetting the CA50 on page 4-17*).
Warm Boot	Warm boot the device (see *Resetting the CA50 on page 4-17*).

Walkie-Talkie (Multicast) Communication

Walkie-talkie communication is initiated and controlled by the UCA client application running on the CA50. This feature operates in speaker phone mode as well as with an earbud. walkie-talkie communication is initiated by pressing a key on the CA50 at which time the device sounds a tone to prompt the user that voice communication can begin. The user is also prompted with an alert tone indicating an incoming communication. Speaker and earbud volume is controlled by the keys on the client device and can be adjusted at any time, except while on a voice (unicast) call (see *Voice (Unicast) Calls on page 4-36*).

Walkie-talkie functionality supports up to 32 separate channels that are prioritized by channel number (where 1 is the highest priority and 32 is the lowest). The UCA server and UCA client application can associate channels with functional names, such as emergency, department, etc., but the implementation below the UCA client application only supports channel numbers when processing calls.

Once the CA50 device has successfully logged in, the UCA server is not required for the CA50 to function properly in walkie-talkie mode. If the CA50 loses its login information, the CA50 requires an up and running UCA server to log in again to enable walkie-talkie functionality. For additional information, see *CA50 Capabilities with an Offline UCA Server on page 6-58*.

The following basic functionality is present in walkie-talkie mode:

> ***NOTE*** For a detailed overview of walkie-talkie functionality, see *Walkie-Talkie (Multicast) Functionality on page 1-3*.

- A list of channels is provided to the UCA client when a user logs in. For details about how to set channels, see *XML Profile for Configuration on page 7-1*.
- Only one voice at a time can be transmitted half duplex on a channel (i.e., the device hears only one channel at a time; there is no channel mixing).
- In *Normal* mode or *Scan* mode the channel with the highest priority plays.
- In *Single Channel* mode only the voice on the selected channel is heard. All other channels are filtered.
- Walkie-talkie communication has a lower priority than voice (unicast) calls.
  - Walkie-talkie communication cannot interrupt a voice (unicast) call.
  - The voice (unicast) call overrides walkie-talkie communication.
- When a voice (unicast) call interrupts walkie-talkie communication, the device returns to monitoring walkie-talkie traffic after the voice (unicast) call ends.

RF Network Optimization for Walkie-Talkie Communication

- The recommended RF switch (wireless switch/AP) configuration is for a DTIM interval of 200 ms.
- Must arrange propagation of CA50 multicast packets to the entire coverage area.
  - Simplest case - single subnet.
- Configure the multicast queue length at the AP large enough to accommodate the walkie-talkie communication load.
- Turn off multicast mask.
- Lower required data rates.
- Recommend 1 MBps and 2 MBps as the required data rates. No other data rate values should be set as “required data rates.”

Walkie-Talkie Functions

Channel Settings

There are three walkie-talkie channel settings:

- Listen/Talk Channel
  - Users can listen and talk on Listen/Talk channels.
  - A channel can be set to Listen/Talk by setting the *type* to '*T*'; in the user profile under the *WTT* > *Chlist* > *Ch* element.
  - The default channel is also Listen/Talk. A default channel is the channel used in normal mode and it is the channel that a user uses to talk. A channel can be set as a default channel by setting the *type* to '*Def'* in the *WTT* > *Chlist* > *Ch* element.

NOTE One Listen/Talk channel must be designated as the default channel for normal mode walkie-talkie communications.

- Listen Only Channel
  - Users can listen to but cannot talk on Listen Only channels.
  - Users cannot manually select and enter single channel mode on Listen Only channels.
  - A channel can be set to listen only by setting the *type* to '*LO*' in the *WTT* > *Chlist* > *Ch* element.
- Accessible (Non-listen) Channel
  - By default, users cannot hear anything on this channel. The channel is used as a reserve channel for single channel mode.
  - A user must select this channel manually to access this channel.
  - Selecting this channel puts a user in *Single Channel* mode.
  - When selected in *Single Channel* mode, users can talk and listen only on this channel.
  - A channel can be set as *Accessible* (single channel only) by setting the *type* to '*SCO*' in the *WTT* > *Chlist* > *Ch* element.

Device Modes

There are four walkie-talkie communication modes (for additional information, see *page 7-5*). These different modes tell the device how to handle the channels to which it has access. The modes are:

- Single Channel Mode
  - Users can talk and listen only on this channel.
  - The only way to end *Single Channel* mode is to leave the channel manually.
- Normal Mode
  - In *Normal* mode users listen on Listen/Talk channels and Listen Only channels.
  - Users talk on the default channel only.

    NOTE One Listen/Talk channel must be set as the default channel.

  - *Normal* mode is exited when the user manually selects a channel other than the default talk channel, causing the device to enter *Single Channel* mode.
- Scan Mode
  - In *Scan* mode users listen on Listen/Talk channels and Listen Only channels.
  - When a user presses **Talk**, the mode switches to *Single Channel* mode on the last Listen/Talk channel heard.
- Do not disturb Mode
  - *Do not disturb* mode supports an extended period of time when a user wants/needs to be unreachable but is still able to use the business application. This may be required when a user takes a break or is given a task that does not require the communication feature.
  - In *Do not disturb* mode the device does not receive any notifications or incoming walkie-talkie communication.

Naming Walkie-Talkie Channels

There are two options for naming walkie-talkie channels:

- Default naming: By default, all channels are numbered sequentially up to 32 channels (Channel 1, Channel 2, Channel 3...Channel 32).
- Manual naming: Channel names can be manually changed in the server application (for details, see *page 7-24*). Names can include alpha, numeric and special characters (-_.,$#*~) and are limited to a maximum of 14-16 characters. An example of channel naming in store might include channels for a sales associate, front-end and open channels. Channel names are not displayed in the status bar; only the channel number displays.

Operating Walkie-Talkie

Initiating Walkie-Talkie Communication

The user can initiate walkie-talkie communication on any screen when not in a voice (unicast) call or in *Do not disturb* mode.

1. Press and hold the **Walkie-Talkie** key (see *CA50 - Front/Back Views on page 3-2*).
2. A short walkie-talkie start tone sounds.
3. Begin speaking into the microphone.
4. To end the call, release the **Walkie-Talkie** key.

Receiving Walkie-Talkie Communication

The user can receive walkie-talkie communication on any screen except when on a voice (unicast) call or in *Do not disturb* mode.

A start tone sounds on the device when the device detects incoming walkie-talkie communication and a message plays through the speaker. To respond, the user presses the **Walkie-Talkie** key and speaks through the microphone on the device.

Changing the Transmit Channel Using the CA50 Menu

The CA50 desktop displays the default transmit channel. To change the channel, users can access the walkie-talkie channel menu directly from a business application, or the desktop.

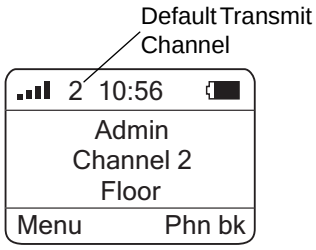


Figure 4-13 *CA50 Desktop*

To change the transmit channel:

1. On the CA50 desktop (*Figure 4-13*), press the left soft key to display the *Desktop* menu.

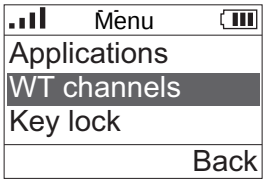


Figure 4-14 *Desktop Menu*

2. Press the **Up** or **Down** keys to scroll and **Select** walkie-talkie communication channels.
   The *WT Channels* menu displays.

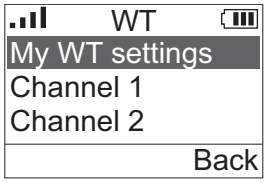


Figure 4-15 *WT Channel Menu*

   The *Channels* menu lists an option to access walkie-talkie settings (*My WT settings*), all available transmit channels and possibly additional server configurable options, such as *Scan mode*.

3. Press the **Up** or **Down** keys to scroll and **Select** a channel.
   The *Desktop* menu (or prior business application screen) displays the updated status bar showing the current receive channel.

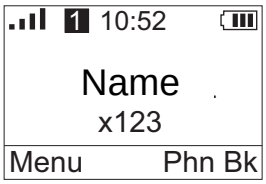


Figure 4-16 *Desktop Menu*

Voice (Unicast) Calls

Voice (unicast) calls are initiated and controlled from the UCA client application running on the CA50 device. The VoIP capabilities provide a means for initiating the call via an IP address or extension, depending on the protocol chosen, and if the system set up requires PBX support.

Voice (unicast) calls are treated as a higher priority than walkie-talkie (multicast) communication. At initial login, availability of the UCA server is required for voice (unicast) calls to function. For additional information, see *CA50 Capabilities with an Offline UCA Server on page 6-58*.

NOTE When the phone book is previously viewed from a CA50, it caches the last accessed phone book information. If the UCA server is temporarily not accessible, then CA50 can access this cached phone book information within the device itself.

Volume control of the output of a voice call is performed by a key press and can only be adjusted while in a voice call. See *Client Settings on page 9-40* for volume keys functionality.

NOTE For a detailed overview of voice (unicast) calls, see *Voice Call (Unicast) Functionality on page 1-3*.

Communication Hierarchy

On Voice (Unicast) Call

When a user is on a voice (unicast) call:

- Walkie-talkie does not function. Incoming walkie-talkie voice is not played and **Walkie-Talkie** keys are inactive.
- When walkie-talkie scan mode is active before a voice (unicast) call, it is automatically re-enabled after the voice (unicast) call ends.
- When walkie-talkie single channel mode is active before the voice (unicast) call, it is automatically re-enabled after the voice (unicast) call ends.
- Business or other application notifications play based on the user profile configuration. Notification tones play through the receiver. Any screen data related to the notification is displayed on the screen.
- A second or additional incoming voice (unicast) calls do not come through and the caller receives a busy signal when no PBX is involved. If a PBX is so configured, the second call could go to voice mail.

Off Voice (Unicast) Call

- Incoming walkie-talkie activities play.
- Incoming business or other application notifications play.

Standard (Non-PBX) Calls

The CA50 device may make voice (unicast) calls to other CA50 devices without the use of a PBX. These are direct CA50-to-CA50 IP based calls.

The UCA server is required to place a call from a CA50. In order to place a call, the CA50 must use a dynamically accessed phone book located on the UCA server. The UCA server dynamically managed phone book downloaded at the initiation of each voice call displays only logged-on users.

NOTE When the phone book is previously viewed from a CA50, it caches the last accessed phone book information. If the UCA server is temporarily not accessible, then CA50 can access this cached phone book information within the device itself.

PBX Calls

The integration of an IP PBX into a CA50 system allows customer calls to a retailer to be forwarded directly to a CA50. In addition, this enables associates using a CA50 to call traditional land line phones through the IP PBX, assuming the land line extension was added to the UCA server dynamically managed phone book.

NOTE 1. The CA50 user interface provides no means for a user to dial an extension via numeric keys. Instead dialing all CA50 calls, including those to an IP PBX accessible standard land line phone extension, must occur via the UCA server dynamically managed phone book.

2. The CA50 cannot be configured for mixed voice call use modes. The CA50 site solution only allows voice calls to be made through either direct CA50-to-CA50 IP based call capabilities (standard non-PBX calls), or through a PBX, but not both within a store implementation. The H.323 PBX mode and standard non-PBX call mode are exclusive.

IMPORTANT If the PBX goes offline, calls to and/or from land lines via the PBX will not work. The UCA system can be configured to work in direct CA50-to-CA50 IP based call or PBX mode. If the UCA system is configured to use a PBX and the PBX goes down, CA50 devices can not make or receive phone calls. This includes CA50-to-CA50 calls.

The UCA server dynamically managed phone book downloaded at the initiation of each voice call displays only logged in users. Additional phone book entries (see *Additional Phonebook Entries on page 9-14*), those not UCA-based (e.g., a wired land line phone), are always displayed.

The UCA server supports the following IP PBX protocols:

- H.323
- Cisco's SCCP (Skinny Call Control Protocol) for integration of Cisco Cell Manager, versions 4 and 5.
- Avaya's SIP (Session Initiated Protocol)

The following H.323 IP PBX is certified to work with the CA50:

- Avaya Communication Manager 5.0 using the H.323 protocol

The following IP PBXs have been tested to work with the CA50:

- Avaya SIP support though the Avaya SIP Enablement Service (SES) version 5.0
- Avaya H.323 Distributed Office V 1.1.2 and 1.2
- Cisco Call Manager Version 5.1

Is the UCA server required to enable a CA50 to communicate through an IP PBX? The answer depends on where the call originates.

- The IP PBX does not require UCA server involvement to route a call to a CA50.

> ***NOTE*** The specifics of the IP PBX are entered into the UCA server so they can be downloaded and programmed into a CA50 at login, enabling the CA50 to contact and register with the IP PBX. After registration is complete, a PBX based call can be routed to a CA50.

- The UCA server is required to place a call from a CA50. In order to place a call, the CA50 must access and use a dynamically accessed phone book located on the UCA server.

> ***NOTE*** When the phone book is previously viewed from a CA50, any portion of the phone book viewed from the CA50, and only this portion of the phone book, is cached on the CA50. If the UCA server is temporarily not accessible, the CA50 can access the cached phone book information.

IP PBX Integration with the UCA Server

The site IT administrator configures the UCA server with PBX information (IP address, available extensions, etc.) enabling CA50 voice calls to be routed to and from the IP PBX.

The following settings must be entered into the UCA server to work with the IP PBX:

- Phone.Gateway.IPAddress
- Phone.Gateway.Port (note the phone Gateway port for H.323 is 1719 (port 5060 for SIP and port 2000 for SCCP)
- Phone.Gateway. Protocol Type
- Phone.Peer2Peer.Enable (port 1720 for H.323 and port 5060 for SIP)
- Phone.Peer2Peer.Port (note the peer-to-peer; direct CA50-to-CA50 IP based call; port 1720 for H.323 and port 5060 for SIP)
- Phone.Gateway.CallPark.Enable
- Phone.Gateway.FAC.CallPark
- Phone.Gateway.FAC.CallPick

For details and definitions of each of these parameters, see *Table 6-2 on page 6-48*. For details on the entering phone settings into the UCA server Web Console, see *page 9-41*.

Additionally, to enable a CA50 to access phone extensions contained within the IP PBX, they must be mapped into the UCA server. See *Creating a UCA Phone Extension on page 9-9* for more details.

Losing a Connection Between the CA50 and IP PBX

When the connection between a CA50 and IP PBX is lost a *No Phone* error message displays on the CA50. Users in *No Phone* state do not appear in the Phonebook of other logged in users.

The connection loss can occur under the following circumstances:

- Temporary network related issues. In this case the connection loss is transient.
- Prolonged network issues.
- IP PBX is down.

No Phone Error Message

A *No Phone* error message displays on the CA50 under the following circumstances:

- A lost connection between the CA50 and IP PBX.
- The phone subsystem gets an unexpected extension from the IP PBX. For example, the IP PBX returns an extension based on the SCCP MAC address.
- When the phone subsystem is not able to register an extension with the IP PBX. This can occur due to an:
  - incorrect IP address
  - incorrect port of PBX
  - invalid extension
  - incorrect password.
- No free extension is available to be assigned.

Typically when the contact list returns to the CA50, the extension number displays along with the user name. For users with no phone extensions, only the user name displays.

Importing and Creating Extensions

For details see *page 9-13* and *page 11-14*.

Voice (Unicast) Call Functionality - PBX Calls vs. Direct CA50-to-CA50 IP Based Calls

Table 4-7 *PBX Calls versus Standard Non-PBSX Calls*

Function	H.323	H.3232:P2P	SIP:PBX	SIP:P2P	Cisco Call Manager
Place Call	• Caller CA50 gets a call progress tone from PBX. • Caller CA50 displays the outgoing caller ID.	• Caller CA50 plays a call progress tone locally. • Caller CA50 displays the outgoing caller ID.	• Caller CA50 gets a call progress tone from PBX. • Caller CA50 displays the outgoing caller ID.	• Caller CA50 plays a call progress tone locally. • Caller CA50 displays the outgoing caller ID.	• Caller CA50 gets a call progress tone from PBX. • Caller CA50 displays the outgoing caller ID.
Receive Call	• Called CA50 plays a ring tone locally. • Called CA50 displays the caller's caller ID.	• Called CA50 plays a ring tone locally. • Called CA50 displays the caller's caller ID.	• Called CA50 plays a ring tone locally. • Called CA50 displays the caller's caller ID.	• Called CA50 plays a ring tone locally. • Called CA50 displays the caller's caller ID.	• Called CA50 plays a ring tone locally. • Called CA50 displays the caller's caller ID.
Call Hold	• If the CA50 puts the call on hold, no music-on-hold is heard on either end. • If Avaya phone puts the CA50 on hold, music-on-hold is heard on both ends.	• No music-on-hold heard on either end.	• If the CA50 puts the call on hold, music-on-hold is heard on both ends (if the PBX is setup with music-on-hold). • If Avaya phone puts the CA50 on hold, music-on-hold is heard on both ends.	• No music-on-hold heard on either end.	• If the CA50 puts the call on hold, music-on-hold is heard on both ends (if the PBX is setup with music-on-hold) • If Cisco phone puts the CA50 on hold, music-on-hold is heard on both ends (if the PBX is setup with (music-on-hold).

Table 4-7 *PBX Calls versus Standard Non-PBSX Calls (Continued)*

Function	H.323	H.3232:P2P	SIP:PBX	SIP:P2P	Cisco Call Manager
Calling a CA50 Already on a Call	• Caller gets a call progress tone from PBX. • If called account is setup with coverage path on the PBX, the call is forwarded to the coverage path. • If the called CA50 keeps the current call, caller hears the call progress tone until timeout setup on PBX, or caller places the phone on the hook. • If the called CA50 hangs up the current call, it can answer this call.	• Caller hears a busy tone.	• Caller gets a call progress tone from PBX. • If called account is setup with coverage path on the PBX, the call is forwarded to the coverage path. • If the called CA50 keeps the current call, caller hears the call progress tone until timeout setup on PBX, or caller places the phone on the hook. • If the called CA50 hangs up the current call, it can answer the new call.	• Caller hears a busy tone.	• Caller gets a call progress tone from PBX. • If called account is setup with coverage path on the PBX, the call is forwarded to the coverage path. • If the called CA50 keeps the current call, caller hears the call progress tone until timeout setup on PBX or the caller disconnects the call.
Call Transfer	• One step (blind) transfer. • If the call is transferred between CA50s, both ends of the CA50 show the transferee caller ID and not the actual caller ID.	N/A	N/A	• One step (blind) transfer. • If the call is transferred between CA50s, both ends of the CA50 show the updated actual caller ID.	• One step (blind) transfer. • If the call is transferred between CA50s, both ends of the CA50 show the updated actual caller ID.

Table 4-7 *PBX Calls versus Standard Non-PBSX Calls (Continued)*

Function	H.323	H.3232:P2P	SIP:PBX	SIP:P2P	Cisco Call Manager
Call Park / Call Retrieve	• CA50 (whose call was parked) is put on hold. • CA50 who parks call gets prompt tone "du, du, du' from PBX and then auto hang up. • Other CA50 units in the same group/department get vibration notification. User can either pickup the call or dismiss it. • After the parked call is retrieved the caller ID displayed on the screen is ID of the parked number not the actual caller ID.	N/A	N/A	N/A	• CA50 (whose call was parked) is put on hold. • CA50 who parks call gets prompt tone from PBX and then auto hang up. • Other CA50 units in the same group/department get vibration notification. User can either pickup the call or dismiss it. • After retrieved the parked call, the actual caller ID shows on the screen.

Browsing the Phone Book

The phone book provides access to all currently logged in CA50 devices and any and all static or additional extensions.

The phone book is divided into static and dynamic sections. Each dynamic section downloads to the device from the server each time the CA50 is used. Dynamic sections are small and do not take a significant amount of time to download (no more than a typical screen download). Static sections download at CA50 login and can be updated if the server configuration options change.

By default, the *Phone Book* menu includes the following (server configurable) options: Extensions, First Names, Last Names, Departments and Recent Calls. *Extension* lists all currently logged in extensions and any and all static extensions. *Last Names* lists all currently logged in users by last name.

On any screen in the Phone Book, press the right soft key to go back to the previous screen.

To browse the phone book:

- On the CA50 desktop (*Figure 4-17* A), press the right soft key to display the *Phone Book Menu* (*Figure 4-17* B).

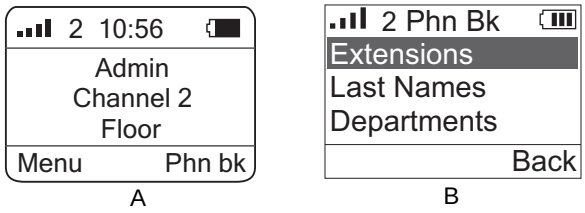


Figure 4-17 *CA50 Desktop (A) / Phone Book Menu (B)*

By Extensions

1. To view extensions in the phone book, press the **Up** or **Down** keys to scroll the *Phone Book* menu (*Figure 4-17 B*), highlight *Extensions* and press **Select** to pick an extension.
   The *Extensions* menu displays a scrollable list of static ranges of currently logged in extensions. Each range contains a maximum of approximately 25 extensions.

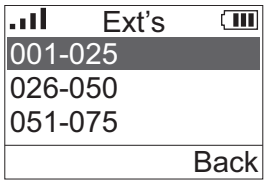


Figure 4-18 *Extensions Menu*

2. Press the **Up** or **Down** keys to scroll and **Select** an extension range.
   The *Extension List* (for the selected range of extensions) displays.

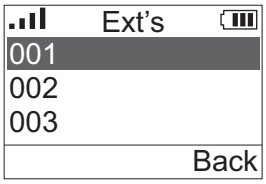


Figure 4-19 *Desktop Menu*

3. Press the **Up** or **Down** keys to scroll and **Select** the extension to call. A voice (unicast) call connects (see *Voice (Unicast) Calls on page 4-36*).

By Last Names

1. To view last names in the phone book, press the **Up** or **Down** keys to scroll the *Phone Book* menu (*Figure 4-17* B) and **Select** *Last Names*.
   The *Letter Range* menu displays a scrollable list of letters, alphabetical ranges representing the last names of currently logged in users. Each range contains a maximum of approximately 25 logged in users.

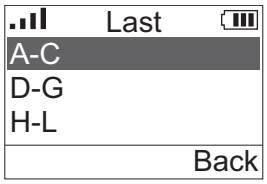


Figure 4-20 *Letter Range Menu*

2. Press the **Up** or **Down** keys to scroll and **Select** a range of letters.
   The *Last Names* menu displays the names of logged in users, alphabetically, in the range selected.

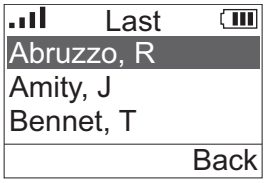


Figure 4-21 *Last Names Menu*

3. Press the **Up** or **Down** keys to scroll and **Select** the name to call. A voice (unicast) call connects (see *Voice (Unicast) Calls on page 4-36*).

By Departments

1. To view departments in the phone book, press the **Up** or **Down** keys to scroll the *Phone Book* menu (*Figure 4-17* B) and **Select** *Last Names*.
   The *Departments* menu displays an alphabetical scrollable list of Departments in the store with users who are currently logged in.

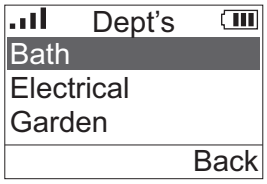


Figure 4-22 *Departments Menu*

2. Press the **Up** or **Down** keys to scroll and **Select** a Department.
   All users currently logged in within the selected department are listed.

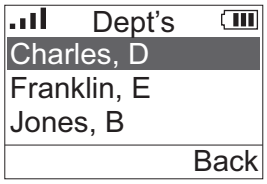


Figure 4-23 *Last Names Menu*

3. Press the **Up** or **Down** keys to scroll and **Select** the name to call. A voice (unicast) call connects (see *Voice (Unicast) Calls on page 4-36*).

Making a Voice (Unicast) Call

To make a call, browse for extensions, names and/or departments in the phone book. The UCA server dynamically managed phone book downloaded at the initiation of each voice call displays only logged in users. Additional phone book entries (see *Additional Phonebook Entries on page 9-14*), those not UCA-based (e.g., a wired land line phone), are always displayed.

The CA50 displays the name of the person called throughout the active call. When the call connects, the caller hears a ringing sound through the receiver. While the call is in progress, the CA50 displays the caller ID and a call timer. Call options, such as *Hold* and *Transfer*, are available via soft keys. When the call ends the device returns to the desktop or prior business application screen (see *Sample Business Application on page 4-54*).

To make a voice (unicast) call:

1. Browse the phone book to find the person to call (see *Browsing the Phone Book on page 4-42*).
2. Press the **Up** or **Down** keys to scroll and press **Select** on the extension, name or department to call. A voice (unicast) call connects (see *Voice (Unicast) Calls on page 4-36*).

Receiving a Voice (Unicast) Call

The device can receive a call on any screen. By default, a voice (unicast) call overrides all other device operations. For information about voice (unicast) calls while in business application see *Sample Business Application on page 4-54*.

When a call comes in, a ring tone sounds and green LED blinks.

On an incoming call, the caller ID displays.

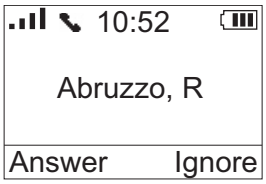


Figure 4-24 *Incoming Call Display*

Press the left soft key to answer the call, or press the right soft key to ignore the call and return to the prior screen.

When the call is answered, the active call screen displays.

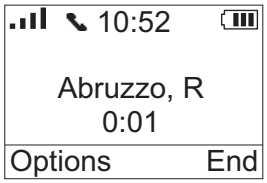


Figure 4-25 *Active Call Display*

During the call, press the left soft key to display options such as *Hold* and *Transfer*.

For information about voice (unicast) calls while in business application see *Sample Business Application on page 4-54*.

Transferring a Call

When a user is on a voice (unicast) call and needs to transfer the call to another extension, use the phone book to look up extensions and transfer the call as shown below.

To transfer a voice (unicast) call:

1. On an active call, *Active Call* window displays.

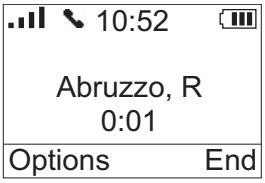


Figure 4-26 *Active Call Display*

2. On the *Active Call* display, press the left soft key to display the *Call Options* menu.

Figure 4-27 *Call Options Display*

3. Press the **Up** or **Down** keys to scroll to *Transfer* and press **Select** to display the *Phone Book* menu.

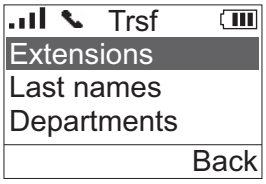


Figure 4-28 *Phone Book Display*

4. Press the **Up** or **Down** keys to scroll to any of the phone book options (*Extensions, Last names or Departments*). Each option allows the user to choose a target to which the call can be transferred (see *Browsing the Phone Book on page 4-42*). In this example *Extensions* is chosen. Press **Select** to display the top level list of all extensions ranges in the phone book.

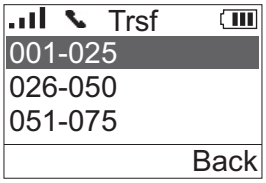


Figure 4-29 *Extensions Range Display*

5. Press the **Up** or **Down** keys to scroll to the appropriate extensions range and press **Select** to display the list of all currently logged in extensions

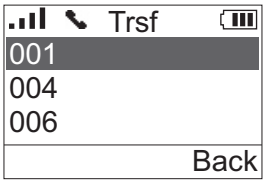


Figure 4-30 *Extensions List Display*

6. Press the **Up** or **Down** keys to scroll to the appropriate extension and press **Select** to transfer the call.

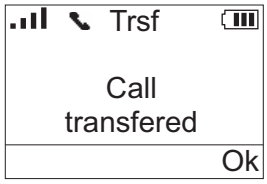


Figure 4-31 *Call Transferred Display*

7. When the call is transferred to the selected extension, it appears on the target device as an incoming voice (unicast) call.

8. Press the right soft key (*Ok*) to return to the desktop or to the prior business application screen.

NOTE When the device detects no user activity after 3 seconds, the device defaults to an *Ok* right soft key press and returns to the desktop, or the prior business application screen.

Parking a Call

When the CA50 receives a voice (unicast) call, the call can be *parked* for retrieval by another CA50 device.

There are two *Call Park* options:

- Parking the call on any available extension allowing the user to pick up the call.
- Parking the call on an available extension within a group and/or department allowing the user to pick up the call.

NOTE The options above are available only if they are enabled on the UCA server. The system variable corresponding to this setting is Phone.Gateway.CallPark.ByGroup.Enable. See *Table 6-2 on page 6-48* for more information. Call park is available only in PBX mode.

Parking a Call on any Available Extension

To park a voice (unicast) call on any available extension:

1. On the *Active Call* window press the left soft key (*Options*) to display the *Call Options* menu.

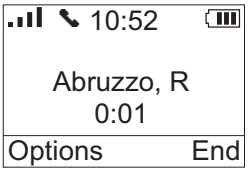


Figure 4-32 *Active Call Display*

2. On the *Call Options* menu press the **Up** or **Down** keys to scroll to *Park* and press **Select**.

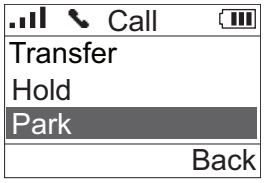


Figure 4-33 *Call Options Display*

3. The system selects an available extension on which to park the call and the *Park* window displays the extension. Press the right soft key (*Ok*) to accept the extension and park the call.

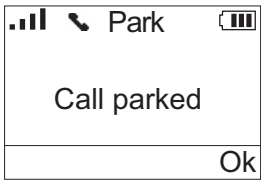


Figure 4-34 *Park Display - On the device on which the call was parked*

NOTE When the user selects *Ok* the device display returns to the desktop or the prior business application. After three seconds of no user action, the device display also returns to the desktop or the prior business application.

4. A message transmits to all other users to inform them that a call is parked on an extension. The user can press the left soft key to pick up the call or the right soft key to dismiss the message.

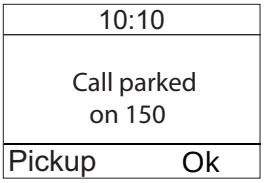


Figure 4-35 *Park Display - On a logged in device other than the device on which the call was parked*

Parking a Call on any Available Extension Within a Group/Department

To park a voice (unicast) call on any available extension within a group or department:

1. On the *Active Call* window press the left soft key (*Options*) to display the *Call Options* menu.

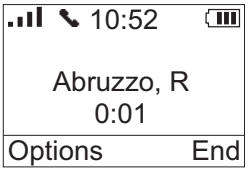


Figure 4-36 *Active Call Display*

2. On the *Call Options* menu press the **Up** or **Down** keys to scroll to *Park* and press **Select**.

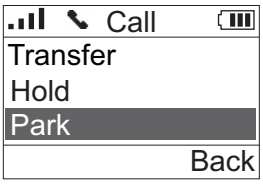


Figure 4-37 *Call Options Menu*

3. The *Departments* menu displays. Press the **Up** or **Down** keys to scroll to the department to which the call should be parked. and press **Select**.

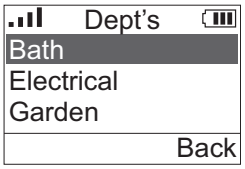


Figure 4-38 *Departments Menu*

4. The system selects an available extension on which to park the call and the *Park* window displays the extension. Press the right soft key (*Ok*) to accept the extension and park the call.

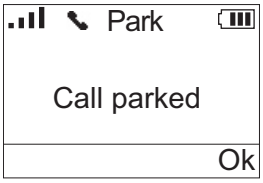


Figure 4-39 *Park Display - On the device on which the call was parked*

NOTE When the user selects *Ok* the device display returns to the desktop or the prior business application. After three seconds of no user action, the device display also returns to the desktop or the prior business application.

5. A message transmits to all other users logged into the same group where the call was parked to inform them that a call is parked on an extension. The user can press the left soft key to pick up the call or the right soft key to dismiss the message.

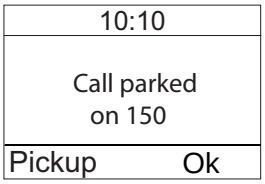


Figure 4-40 *Park Display - On a logged in device other than the device on which the call was parked*

No Extension Available to Park

When no extensions are available to park a voice (unicast) call the display shows an *All extensions taken* message. Press the right soft key (*Back*) to return to the *Call Options* menu (*Figure 4-37*) to try again, or select another option (see *Transferring a Call on page 4-46* and *Hold on page 4-53*).

Retrieving a Parked Call

Retrieving a Parked Call from any Extension

To retrieve a *parked* voice (unicast) call from the desktop:

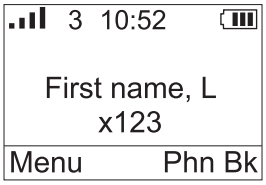


Figure 4-41 *Desktop Display*

1. On the *Desktop* (*Figure 4-41*) press the left soft key (*Menu*) to display the *Desktop* menu.

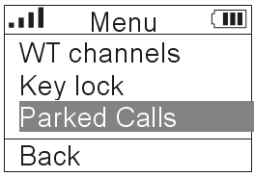


Figure 4-42 *Desktop Menu*

2. Press the **Up** or **Down** keys to scroll to *Parked Calls* and press **Scan/Select (S)**. The *Pick* menu displays. This menu lists all parked extensions. Extension numbers vary based on numbers assigned by a store administrator.

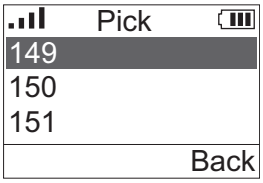


Figure 4-43 *Call Pick Menu*

3. Press the **Up** or **Down** keys to scroll to an extension in the list and press **Scan/Select (S)**. When selected, the parked call on the extension unparks and the *Active Call* displays.

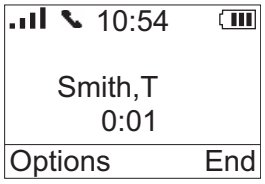


Figure 4-44 *Active Call Display*

NOTE When a parked call is not picked up within the time configured in the PBX, the PBX may transfer the call back to the originating device.

4. Press the right soft key to *End* the call.

Retrieving a Parked Call from an Extension Within a Group/Department

To retrieve a *parked* voice (unicast) call from the desktop:

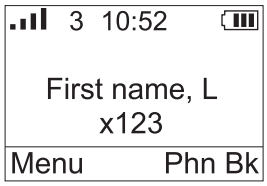


Figure 4-45 *Desktop Display*

1. On the *Desktop* (*Figure 4-45*) press the left soft key (*Menu* > *Options*) to display the *Call Options* menu.

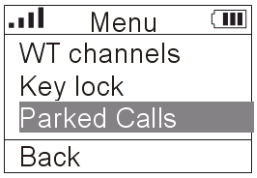


Figure 4-46 *Desktop Menu*

2. Press the **Up** or **Down** keys to scroll to *Pickup* and press **Scan/Select (S)**. The *Dept's* menu displays.

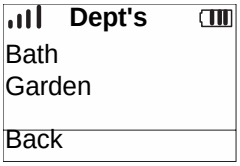


Figure 4-47 *Dept's Menu*

NOTE Only *Groups* with parked calls show in the *Dept's* menu. For example, if calls are parked only in the Bath Group, Bath is the only Group to display. If calls are parked in both the Bath and Garden Groups, both Groups display.

3. On the *Dept's* menu press the **Up** or **Down** keys to scroll to the appropriate *Group* (Department) and press **Scan/Select (S)**. The *Pick* menu displays. This menu lists all parked extensions within the *Group* selected from the *Dept's* menu (*Figure 4-47*). Extension numbers vary based on numbers assigned by a store administrator.

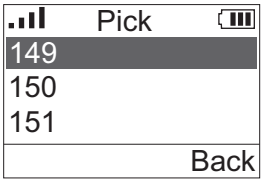


Figure 4-48 *Call Pick Menu*

4. Press the **Up** or **Down** keys to scroll to an extension in the list and press **Scan/Select (S)**. When selected, the parked call on the extension unparks and the *Active Call* displays.

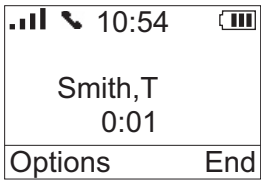


Figure 4-49 *Active Call Display*

NOTE When a parked call is not picked up within the time configured in the PBX, the PBX may transfer the call back to the originating device.

5. Press the right soft key to *End* the call.

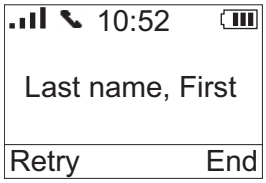
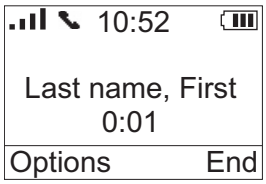
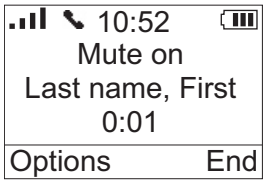
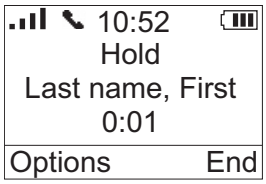
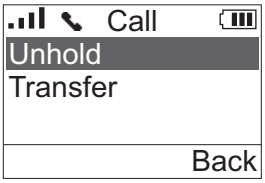
Park/Retrieve Troubleshooting

Table 4-8 *Park Call / Retrieve Call Troubleshooting*

Situation	Description
Caller Terminates Parked Call	If a caller terminates a parked call, the UCA server and the CA50 device still treat the call as a parked call. When the user navigates to pick up the call a fast busy tone sounds on the device from the PBX. The fast busy tone terminates the call on the CA50 and signals the UCA server that the call was picked up. The UCA server removes the extension from the parked calls list.
Calls parked by devices other than the CA50	• Calls parked on the PBX by devices other than a CA50 are not recognized by the UCA server.
Parked call timeout	The timeout duration of parked calls is a configurable parameter on the PBX. When a parked call times out, the call is diverted back to the device that parked the call. • When a parked call times out because the CA50 device is already engaged in conversation, the call does not transfer to the CA50 (call waiting is not supported). • When a parked call times out and the CA50 device is not in conversation mode the parked call is be diverted back to the device that parked the call.
Two different devices attempt to pick up the same parked call	One device picks up the parked call and the other device receives a fast busy tone. The fast busy tone terminates the pick up attempt and the device display returns to desktop.

Voice (Unicast) Call States

Table 4-9 *Voice (Unicast) Call States*

State	Description
Busy Signal 	When caller receives a busy signal when the person called is on another voice (unicast) call. Caller ID and new soft key controls display: • Retry: redials extension • End: returns to prior phone book Item listing.
End Call 	When a call ends when either party presses the right soft key to end the call, the timer stops incrementing. When the call ends the CA50 returns to the desktop or prior business application screen (see *Sample Business Application on page 4-54*).
Mute On 	When the user presses the **Mute** (see *Mute Key on page 4-7*), the microphone is muted. While muted, the call timer continues uninterrupted. Mute is in place until the user presses **Mute** again, or the call ends. The mute state does not carry across separate calls. By default, mute is off at the start of a voice (unicast) call. During a voice (unicast) call, **Mute** is a toggle switch, muting and unmuting the microphone for the duration of a single call. See *Answering an Incoming Voice (Unicast) on page 4-55* for information about receiving a voice (unicast) call while in a business application.
Hold 	When a user selects the hold option while on an active call, the call is placed on hold. While on hold, the call timer continues uninterrupted. See *Answering an Incoming Voice (Unicast) on page 4-55* for information about receiving a voice (unicast) call while in a business application. Press the left soft key to display Options:  • Select *Unhold/Hold* to place the call off and on hold and return to the active call screen. • Select *Transfer* to transfer a call to another party (see *Transferring a Call on page 4-46*).

Sample Business Application

NOTE For additional sample business applications see *Appendix I, Business Applications Supplement*.

By default, pressing the **Scan** key at the desktop launches the scanner and delivers decoded data to the sample item look up application. If the item lookup is successful, data corresponding to the bar code (such as the product type/name and price) is delivered back to the device and displayed (see *Figure 4-50*). If the item lookup is unsuccessful, the server returns an *Item Not Found* message and the decoded bar code content to the device for display (see *Figure 4-51*).

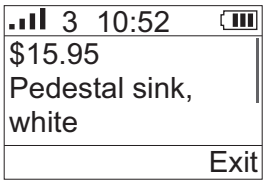


Figure 4-50 *Successful Decode/Item Lookup*

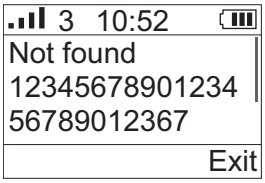


Figure 4-51 *Unsuccessful Decode/Item Lookup*

Users can perform scan functions at the desktop, in a business application or while on a voice (unicast) call (*Voice (Unicast) Calls on page 4-36*).

Accessing the Menu and Phone Book in the Business Application

If the business application uses an *Exit* option via soft keys, press the appropriate soft key to exit the business application and return to the desktop.

The business application developer can modify the XML profiles to provide soft key options for *Menu* and *Phn Bk* for the user to access these options.

The business application developer can use a meta refresh HTML tag to send the application back to a prior screen after a period of time specified in the tag.

Answering an Incoming Voice (Unicast)

Call in a Business Application

NOTE The following steps include instructions to answer a voice (unicast) call from an items detail screen that is part of the sample item lookup application. These instructions apply to any business application screen.

To answer a voice (unicast) call from a business application:

1. While in the business application, the device rings and the green LED blinks, indicating an incoming call.

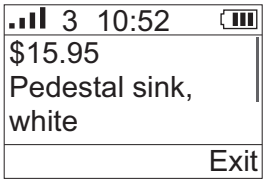


Figure 4-52 *Business Application/Item Lookup*

2. The caller ID displays.

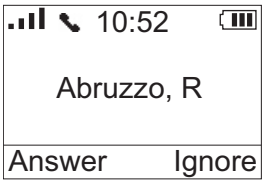


Figure 4-53 *Caller ID*

3. Press the right soft key to dismiss the call and return to the prior business application screen.

   or

   Press the left soft key to answer the call. The active call screen displays.

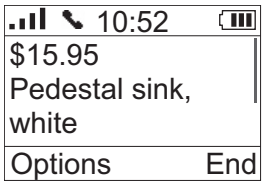


Figure 4-54 *Active Call Screen*

   By default, call control soft keys display *Options* and *End*. Soft key options are server configurable.

For more information about answering a voice (unicast) call, see *Receiving a Voice (Unicast) Call on page 4-45*.

Launching the Sample Business Application While in a Voice (Unicast) Call

While on an active voice (unicast) call, a user may be required to access a business application to acquire information for the call.

NOTE The steps that follow assume that the user was not previously in a business application.

To access the sample business application while on an active voice (unicast) call:

1. On the *Active Call* screen, go to Options > Applications > [Sample business application], aim the device scan window at a bar code and press **Scan**.

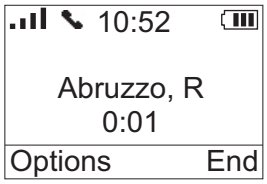


Figure 4-55 *Active Call Display*

2. The *Item Detail* screen displays the scanned product detail.

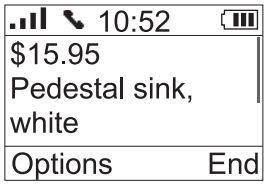


Figure 4-56 *Item Detail Display*

By default, call control soft keys (*Options* and *End*) display. Soft key options are server configurable.

To return to the active (unicast) call:

1. If the business application uses the *Back* soft key option, press the appropriate *Back* soft key to return to the prior screens (active unicast call screen).

   or

2. A business application developer can use a meta refresh HTML tag to send the application back to a prior screen after a period of time specified in the tag.

Accessing Communication Options from a Business Application

Initiating Walkie-Talkie Communication

By default, walkie-talkie communication can be initiated at any time while in a business application by pressing the **Walkie-Talkie** key(s).

NOTE If the application developer changes the function of the **Walkie-Talkie** key(s), this option may not be available.

Accessing the Menu or Phone Book

There are two options to access the menu and/or phone book while in a business application.

- Exit the Business Application - Use one of the soft keys to select *Back* or *Exit*. (In the sample item lookup application, the user must return to the desktop by selecting *Back* in order to access the menu or phone book.)

  or

- Customize the Business Application Soft Keys - Application developers can include links to communication related options within the soft key functionality.

  Example 1:

  A business application screen displays with the soft key options from the desktop; the soft key options function the same as on the desktop.

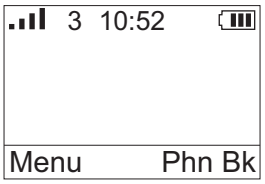


Figure 4-57 *Business Application - Use Desktop Soft Keys*

  Example 2:

  A business application screen displays. The soft key *Options* function displays a menu and the soft key *Back* function navigates to the previous screen and can exit the application.

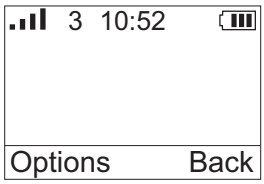


Figure 4-58 *Business Application - Use an Options Menu*

  The *Options* menu contains a link to additional business application functionality (e.g., Item locator) and links to the phone book and the walkie-talkie channel menu.

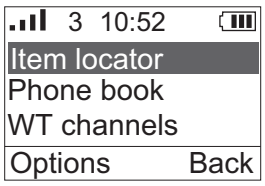


Figure 4-59 *Options Menu - Use an Options Menu*

(Page intentionally blank.)

Chapter 5 UCA Client Application on the CA50

UCA Client Application Overview

The UCA client application which resides and runs on the CA50 is the ONLY interface from the CA50 to the UCA server. The UCA client application is an integral part of the CA50 solution and provides all the following functionality on the CA50:

- Interface for Communication between the CA50 and the UCA Server
- Walkie-Talkie Functionality
- Voice Call Functionality
- Web Browser for running Business Applications
- Employee Identity Validation
  - Scanning employee login bar code
  - Entering employee unique PIN

The UCA client application which resides and runs on the CA50 generates the user interface displayed on the device and is the sole means for end users to access the CA50 device functionality, such as business applications, messages/pages and Voice-over-IP functionality. Out-of-box, the UCA client application supports voice communication (both voice --unicast-- calls and walkie-talkie communication) and arbitrates voice and business application priorities.

The UCA client application, shipped from the factory, is the only application that can run on the CA50. Third party applications cannot be loaded directly onto the CA50. Instead, the UCA client application allows a user access to server based third party business applications via an HTTP 1.1 compliant Web browser. A user can interact with the business application via the CA Menu option which accesses the Web browser. Business applications can be hosted on the UCA server or other addressable Web servers in the enterprise.

The CA50 client server architecture is designed to allow the UCA server to control the client behavior on the device without requiring each CA50 keystroke to send a packet to the UCA server. When a CA50 user logs into the UCA server, it provides all the information needed to configure the UCA client application and all peripheral components that reside on the device. The UCA client application also receives static menus that include the walkie-talkie channel options, user preference selection options, etc. For details, see *Chapter 7, XML Profile for Configuration*

When the phone book is previously viewed from the CA50, any portion of the static phone book viewed from the CA50, and only this portion of the phone book, is cached on the CA50. If the UCA server is temporarily not accessible, the CA50 can access the cached phone book information.

The static menus can also refer to dynamic menus that are downloaded from the UCA server on demand via Web service calls.

The UCA client application is designed to get almost all of its information from the UCA server. An exception to this rule is staging information required at the site deployment such as Store Number, UCA server IP address and Language. This staging information is loaded into the CA50 by scanning MSP generated Rapid Deployment bar codes. The UCA client application uses these values to connect the CA50 to the UCA server and go through the logon process. When the user is logged in, the server sends the client profile data containing the user's device configuration, any personal settings and the menus that user can access. Some menus are not downloaded at login time but are only available by making Web service calls. These are especially important for dynamic data, such as phone lists, since those lists contain only extensions that are currently logged in.

NOTE The UCA client application cannot be launched from the CA50 CA Menu until a valid UCA server address is loaded into the device. For details about how to load this information onto a CA50, see *Appendix F, Rapid Deployment*. Once a UCA server IP address is loaded, the CA Menu automatically launches the UCA client after a reboot.

NOTE 1. For a complete checklist outlining all the development and deployment activities required to implement a CA50 solution, see the *Development and Deployment Checklist on page 2-1*.

2. For a table listing the most commonly requested topics within this manual, see the Quick Startup Guide on the last page of the manual.

(Page intentionally blank.)

Chapter 6 UCA Server

Overview

This chapter provides information about the UCA server software and the UCA server database.

Valuable topics in this chapter include:

- UCA server overview (*page 6-1*)
- Theory of operation (*page 6-6*)
- Minimum server hardware requirements (*page 6-7*)
- Server software requirements (*page 6-7*)
- UCA server installation (*page 6-9*)
- Uninstalling UCA server software (*page 6-42*)
- Validating successful server installation and performance (*page 6-45*)
- Configuring firewalls (*page 6-47*)
- System settings (*page 6-48*)
- UCA server security (*page 6-54*)
- CA50 capabilities with an offline UCA server (*page 6-58*).

UCA Server Overview

The UCA server software includes the core server components, a Web based console for server setup and maintenance, Demo Solution and Development and Integration Tool Kit.

The UCA server uses a database to store the data for configuring, maintaining and tracking the following functionality:

- Device registration
- Login process
- User Profile Delivery
- Voice Communication Setup and Management
  - Voice (unicast) calls
  - Walkie-talkie communication
- Delivery of text and command to a device
- Metrics collection
- Console for server setup and maintenance.

The CA50 Demo Solution allows a user to easily showcase all major CA50 / UCA server functionality:

- Walkie-talkie communication
- Voice (unicast) calls without a PBX
- Price lookup (business application).

For additional information about the CA50 Demo Solution, see *Appendix E, Self Contained Pre-Built Demo*.

The Development and Integration Tools enable software developers to extend the standard Motorola provided server functionality. The tools includes documentation, reference implementations and sample source code.

IMPORTANT The UCA server is an in-store server, required at every facility/location to manage the CA50s in each facility.

NOTE The demo solution, reference implementations and associated source code are provided as an installation option on the UCA server Installshield. The Developer Guide provides guidelines on how to modify the demo solution and reference implementations. No application development support is provided with the installation of this software.

NOTE 1. For a complete checklist outlining all the development and deployment activities required to implement a CA50 solution, see the *Development and Deployment Checklist on page 2-1*.

2. For a table listing the most commonly requested topics within this manual, see the Quick Startup Guide on the last page of the manual.

NOTE The CA50 Server Software ships with Motorola's standard 90-day software warranty. During the warranty period, Motorola will replace damaged or defective media. Telephone and e-mail access are also provided for the sole purpose of reporting a non-conformance issue, and for Motorola to verify the existence of the issue and determination of the conditions under which the issue may recur.

For enhanced support over the standard warranty, including full access to technical support resources and rights to download entitled software releases, Motorola recommends purchasing a CA50 Server Software Support agreement.

To maintain MSP support beyond the 90-day warranty period, Motorola recommends purchasing an MSP Software Support agreement. Note: MSP Software Support agreements must be purchased separately or in addition to CA50 Server Software Support agreements

Server Functionality

The UCA server software uses a database to store the data for configuring, maintaining and tracking the functionality.

IMPORTANT Before running the server, ensure that any firewall(s) between the server component and the CA50 are appropriately open to allow CA50 devices to communicate with the server and receive communications from the server.

Device Registration

The CA50 client registers with the UCA server and receives the initial commands required to begin the login process.

Login Process

An associate identifies him/herself to the UCA server by scanning a user identification bar code and is authenticated by the server for CA50 use within a given role and group.

User Profile Delivery

The server uses the associate's role and group to select a profile template, assigns a phone extension, retrieves user preferences and then fills in and delivers the completed profile to the CA50.

NOTE A profile configures a CA50 with authorization levels and privilege rights by setting the user identification, phone configuration, walkie-talkie configuration, UI appearance and navigation, and device configuration settings.

Voice Communication Setup and Maintenance

Voice (Unicast) Calls

To enable dialing without a keypad, the server maintains a dynamic phone book of extensions that are delivered to the CA50 upon request. Voice (unicast) calls are placed and transferred from the phone book via menu options.

Walkie-Talkie Communication

When a profile that specifies walkie-talkie channels is downloaded to a CA50, no further UCA server interaction is required to enable walkie-talkie communication. The phone book is not used when communicating via walkie-talkie mode.

Delivery of Text and Commands to a Device

Enables text and commands from external and system sources to be routed to CA50s.

Metrics Collection

The server can be configured to collect CA50 usage metrics such as the following: number of walkie-talkie communications, voice (unicast) calls and their duration. In addition, battery level, signal strength and time in *Do not disturb* mode are also collected.

Console for UCA Server Setup and Maintenance

A Web based console is provided for server setup and maintenance.

Demo Solution

The CA50 demo solution allows a user to easily showcase all the major CA50/UCA server functionality:

- Walkie-talkie communication
- Voice (unicast) calls without a PBX
- Price lookup (business application)
- HRA (business application)
- Stock check (business application)
- Item locationing (business application).

This demo includes all the documentation and bar codes required to configure and demonstrate the functionality above.

For developers, the demo solution is a bundling of capabilities such as a price lookup reference implementation (business application) and sample server configuration. The demo solution exposes developers to the following concepts:

- User login
- Defining groups and roles through the UCA WebConsole Server Interface
- Setting up a phone book with extensions through the UCA WebConsole Server Interface
- Dynamically assigning new phone extensions to users as they login
- Asynchronous text messaging.

Business Application and Associated Database

Sample item lookup (price check) is available through a demo item lookup database.

For detailed information about the Demo Solution, see *Appendix E, Self Contained Pre-Built Demo*.

For sample business applications see *Appendix I, Business Applications Supplement*.

Development Toolkit with Sample Source Code

The Development Toolkit is installed in a default location in the following folder:

\Program Files\Motorola\UCA Server\DevelopmentToolkit\.

The toolkit includes various samples comprised of source code with solution/project files for Visual Studio 2005.

- Sample Business Applications - Sample Lookup and Sample Lookup 2
- UCA Messaging Sample (see *page 11-44*)
- UCA Phone Extension Sample (see *page 11-46*)
- Sample Device User Authentication Web Service (see *page 11-8*)
- Sample Console User Authentication Web Service (see *page 11-40*)
- Sample CSV device user and phone extension import files.

Required Development Tools

- Visual Studio 2005 Team Edition for Developers with SP1 installed
- Microsoft Web Deployment Projects installed.

Theory of Operation

The following lists the basic process the CA50 client executes at start up:

- Registration (see *Device Registration on page 6-4*)
- Authentication/login (see *Chapter 11, UCA Server Optional APIs*)
- Profile retrieval (see *Chapter 7, XML Profile for Configuration*)
- Desktop launch (see *CA50 Login - Post-deployment on page 4-2*).

The following sequential steps are an overview of the events that occur when a CA50 powers on, runs the UCA client application and a user logs in:

1. The client calls the DeviceRegistration Web Service method (*page 6-4*) to let the server know that it is present.
2. The server adds the device information to a table for identification and checks if the device is authenticated directly, or if the user must login. If the device is authenticated, the server immediately builds and sends the XML profile to the client (*step 5*). If the device is not authenticated, the server proceeds to the user login process (*step 3*).
3. The server sends back an XML command to browse to a Web page to begin the login process.
4. The client posts back user input while proceeding through the login process.
5. When the login process reaches the point where the server has enough information, it begins building the XML profile to send back to the client.
6. The server uses the user's role and primary group to determine which base profile to use to build the user profile. This relationship, tying the user’s role and group to a specific base profile, is defined by creating a profile association through the UCA server Web Console.
7. The base profiles are located in the following directory in the default installation: \Program Files\Motorola\UCA Server\Profiles. The profiles can be configured on the *UCA Server Console* > *System Settings* > *System.Profile.Path*.
8. Once the base profile is chosen, the server begins modifying the base profile with data specific to the user that is logging in.
9. The server assigns a phone extension to the user based upon his or her primary group and writes it to the profile.
10. The server collects any user preferences specific to the user and writes it to the profile.
11. The resulting profile, which was built from a base profile, is sent (returned) to the CA50 device.
12. The client processes the XML profile and configures the CA50 device.
13. When the configuration is complete, the client displays the desktop. At the desktop the user has several options. Among them are options to launch a business application, place a phone call, change walkie-talkie channels, enter *Do not disturb* mode or set preferences.

UCA Server Requirements and Setup

Minimum Server Hardware Requirements

NOTE Server hardware is not available from Motorola and must be purchased separately.

Recommended UCA server hardware configuration to support up to a 50 device (CA50) deployment:

- CPU: 3 GHz Pentium 4 Processor
- RAM: 2 GB
- Hard drive: Minimum 10 GB free disk space

IMPORTANT The recommended UCA server hardware configuration listed above is an estimate and should be tested based on usage needs.

Server Software Requirements.

The UCA server software must be run on a server operating system that meets the requirements listed below.

Operating System

- Server Operating System:

NOTE The UCA server must be set to a static IP address or a DNS resolvable host name to allow communication from CA50 devices.

- **Demo and Deployment**: Microsoft Windows Server 2003 v.6.0 SP1 (This Operating System is required for site deployments.)
- **Demo only**: Microsoft Windows XP Pro SP2 v.5.1 (This Operating System is for demonstration purposes only.)

NOTE 1. Windows XP Pro lacks a DHCP server.

2. By default Windows XP Pro IIS is limited to 10 connections (mobile devices).

Operating System Components That Must Be Enabled

- Internet Information Services (IIS)
  - Common Files
  - World Wide Web Service
  - IIS
    - IIS Snap-In (Windows XP Pro)
    - IIS Manager (Windows 2003)

NOTE IIS must be installed and running before the installation software runs.

To confirm IIS is enabled and running properly, perform the following steps:

- Go to *Start* > *Settings* > *Control Panel* > *Administrative Tools* > *Services* and select "IIS Admin Service."
- Select the *General* tab on the *IIS Admin Properties* window and ensure Startup Type = Automatic and Service Status = Started.
- Go to *Start* > *Settings* > *Control Panel* > *Administrative Tools* > *Services* and select "World Wide Web Publishing Service.
- Select the *General* tab on the *World Wide Web Publishing Properties* window and ensure Startup Type = Automatic and Service Status = Started.

- Optional Server Applications to Install
  - Microsoft SQL Server Management Studio Express - enables direct viewing of the database without using the Console for Server Setup and Maintenance.

Installation Software

- UCA Server Components - Loaded by the Installation Software
  - Web Applications
    - Web pages
    - SOAP Web services
    - REST Web services
  - Windows Services
  - Various Utility Applications.
- Additional Software - Loaded by the Installation Software
  - SQL Server 2005 Express
  - .NET Framework 2.0.

UCA Server Installation

The UCA server installation process performs the following steps:

1. The UCA server must be set to a static IP address or a DNS resolvable host name to allow communication from CA50 devices.
2. Sets up eight Web applications: Client, UCAConsole, ServerManagement, UCASampleAuth, UCASampleApp, UCASampleApp2, UCAMessaging and UCAPhoneExtensions.
3. Installs Windows services (Motorola UCA Maintenance Service, Medial Relay Service).
4. Installs support utilities: DBExport, Start UCA Services, Stop UCA Services, UDT, Encrypt and Decrypt.
5. Installs sample profile templates files used by the UCA login process as specified in the sample data.
6. Installs a named instance of Microsoft SQL Server 2005 Express.
7. Creates the UCA database and adds system defined data and sample data.
8. Creates a SampleLookup database for the Sample Applications and populates with sample data.
9. Creates three IIS Application pools (Client, UCAConsole and ServerManagement) and configures the corresponding Web applications to use those pools.
10. Sets Basic Authentication for the ServerManagement virtual directory, but not for the Authentication subfolder under it.
11. Creates a customer specified login for the database with a database role allowing it to access the necessary portions of the database for the UCA server.
12. Gives the IIS user group permission to access the export folder under Program Files.
13. Encrypts the Web config and application config files to protect the connection strings and make it more difficult for someone to redirect the authentication Web service calls.
14. Installs Motorola UCA Backup and Restore tool.
15. Installs Motorola UCA Profile Builder.
16. Installs Motorola UCA Diagnostics Tool.
17. Installs resource files for localization support.

Installation Prerequisites

See *Server Software Requirements. on page 6-7*.

Installing UCA Server Software

There are three different versions of the UCA server software installation.

- Production version - Basic UCA server software version without Demo data.

  Production version is meant to be installed on a unit where there was no prior version of UCA installed. If an attempt is made to install the production version on the server where prior version of UCA server is installed, a warning message appears instruction the user to un-install any prior versions of UCA server before installing this version.

  If the production version detects the demo version (version 2.0) of UCA server then it upgrades the demo version to production version. It will not change the database or any of the demo data.

- Demo version - Contains demo data in addition to the basic installation and is for evaluation purposes only. The Demo version can be upgraded to a production version by using the production version install. Demo version is same as production version with additional sample data and sample applications. Sample data includes sample phone extensions, users, groups, etc. Demo version helps with understanding the UCA server usage.

- Upgrade version - Upgrades the prior version of UCA server software (1.4.x.x).

  Upgrade version is meant for upgrade from Rev A (1.4) to Rev B (2.0). In this case one has to take the backup of Rev A before installing the upgrade package. Procedure is as outlined below:

  Creat a backup of Rev A database

  - It generates an archive file.
  - Unzip the archive in some known location.

  Install upgrade version of UCA server.

  Install package would prompt for Database backup. Point to the location where the zip file was un-archived.

NOTE Before installing the upgrade version see *page 6-42*, *Option B: Upgrade the UCA server software to a newer version. This requires only Action 1.*.

UCA server software is available on a CD. The UCA server software bundle ships with two components:

- UCA server installation CD
- UCA server Entitlement Certificate (documents the number of servers on which UCA server software may be loaded).

See *Table 14-1 on page 14-4* for UCA server install/uninstall troubleshooting.

IMPORTANT The installation takes approximately 10 to 25 minutes, depending on whether or not .NET Framework is already loaded. During the installation process, the Installshield progress bar may not always appear to be moving. This is normal behavior and does not mean the installation process stopped.

Installing Production Version

To install the Production version of the UCA server software:

1. Start the *UCA Server Software for the CA50* CD.
2. From the CD main menu, click *Install UCA Server Software for the CA50*.If Microsoft .Net 2.0 Framework is not installed on the host machine the following window displays *Install Microsoft .Net 2.0 Framework* Click **Yes** to install the UCA Server Software.

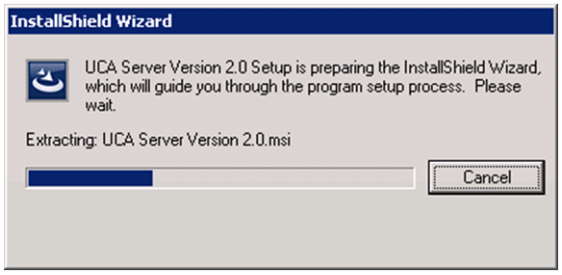

InstallShield Wizard
UCA Server Version 2.0 Setup is preparing the InstallShield Wizard, which will guide you through the program setup process. Please wait.
Extracting: UCA Server Version 2.0.msi
Cancel

Figure 6-1 *Install UCA Server Software*

3. Microsoft .Net 2.0 Framework installs. (This may take several minutes.) When complete, the *Welcome* window displays.

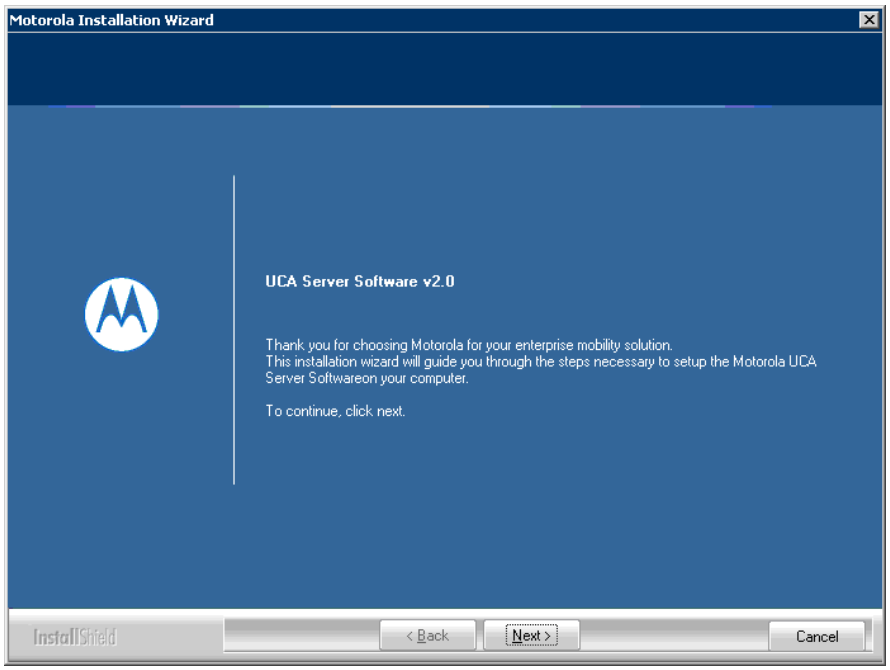

Motorola Installation Wizard
UCA Server Software v2.0
Thank you for choosing Motorola for your enterprise mobility solution.
This installation wizard will guide you through the steps necessary to setup the Motorola UCA Server Softwareon your computer.
To continue, click next.
InstallShield
< Back
Next >
Cancel

Figure 6-2 *Welcome*

4. Click **Next** on the *Welcome* window.
   The *License Agreement* displays.

Motorola Installation Wizard
License Agreement
Please read the following license agreement carefully.
MOTOROLA, INC.
END USER LICENSE AGREEMENT
Motorola is willing to license its Software and the accompanying documentation to you only on the condition that you accept all the terms in this License Agreement ("Agreement").
IMPORTANT: READ THE FOLLOWING TERMS AND CONDITIONS BEFORE USING THE ACCOMPANYING PRODUCTS.
BY CLICKING ON THE "ACCEPT" BUTTON BELOW, YOU ACKNOWLEDGE THAT YOU HAVE READ THIS AGREEMENT, UNDERSTAND IT AND AGREE TO BE BOUND BY THE TERMS OF THIS AGREEMENT. IF YOU DO NOT AGREE TO THE TERMS OF THIS AGREEMENT, MOTOROLA IS NOT WILLING TO LICENSE THE PRODUCTS TO YOU. YOU SHOULD CLICK ON THE "DO NOT ACCEPT" BUTTON TO DISCONTINUE THE SOFTWARE INSTALLATION PROCESS. IF YOU DO NOT AGREE TO THESE TERMS, YOU MAY, WITHIN 15 DAYS, RETURN THIS ENTIRE PRODUCT TO THE LOCATION WHERE YOU ACQUIRED IT OR PROVIDE WRITTEN VERIFICATION OF DELETION OF ALL COPIES OF THE ENTIRE PRODUCT IF YOU HAVE NOT PHYSICALLY RECEIVED A PRODUCT FOR A FULL REFUND.
1. Definitions.
I accept the terms of the license agreement
Print
I do not accept the terms of the license agreement
InstallShield
< Back
Next >
Cancel

Figure 6-3 *License Agreement*

5. Read the license agreement and click **Yes** to accept and continue the installation.
   The *Customer Information* window displays with the current user name and company. Change if necessary.

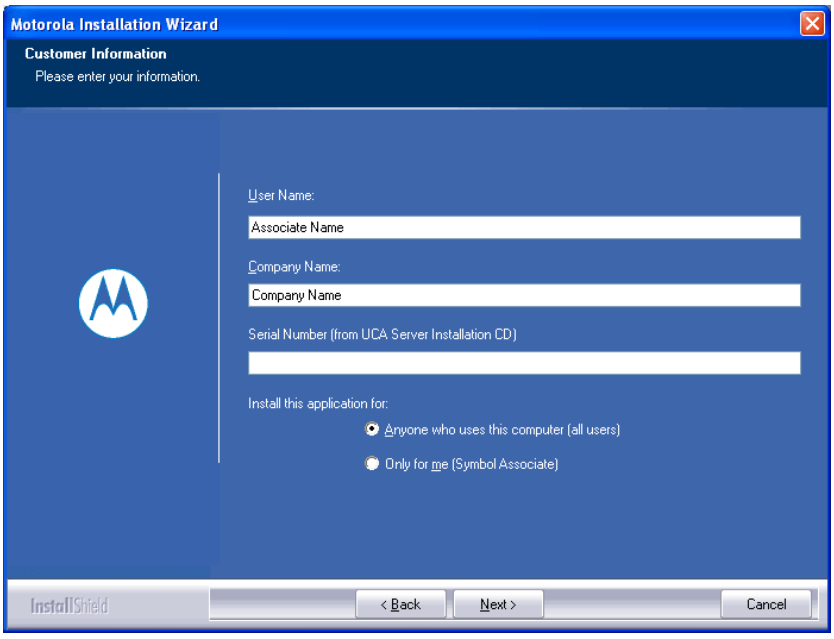


Figure 6-4 *Customer Information*

6. Enter the serial number that appears on the back of the UCA installation CD sleeve.

7. Click **Next**.
   The *SQL Server 2005 Information* window displays.

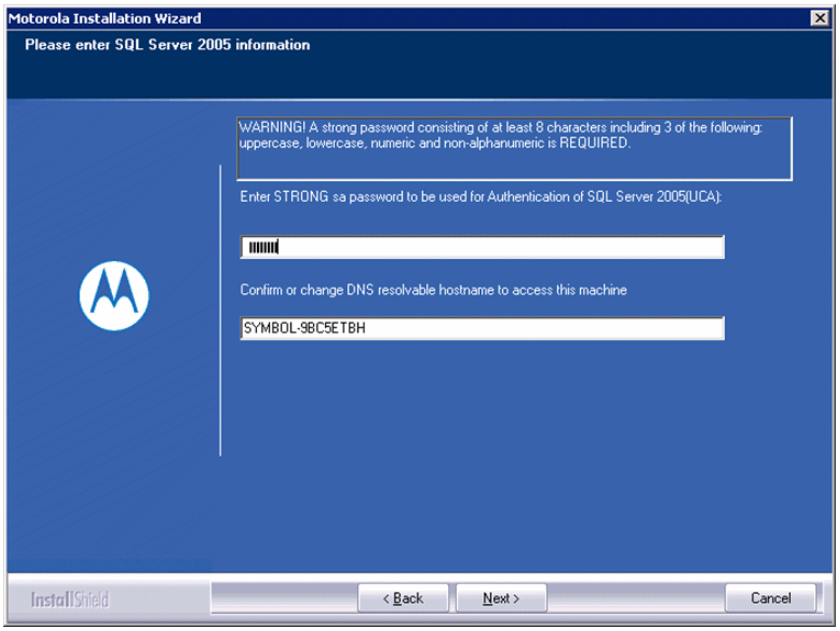

Motorola Installation Wizard
Please enter SQL Server 2005 information
WARNING! A strong password consisting of at least 8 characters including 3 of the following: uppercase, lowercase, numeric and non-alphanumeric is REQUIRED.
Enter STRONG sa password to be used for Authentication of SQL Server 2005(UCA):
Confirm or change DNS resolvable hostname to access this machine
SYMBOL-9BC5ETBH
InstallShield
< Back
Next >
Cancel

Figure 6-5 *SQL Server 2005 Information*

8. Enter a strong password[1] to use for *SA* account.

NOTE The password entered in this field is the system administrator (sa) password. The sa account in SQL Server 2005 has full access and read/write permissions to all DB instances. It is used during the installation process and for future upgrades. This password should be recorded and retained in a safe location by the integrator or customer.

IMPORTANT Microsoft Strong Password Guidelines for SQL Server 2005 authentication:
Strong passwords are not readily guessed by a person, and are not easily hacked using a computer program. Strong passwords cannot use prohibited conditions or terms, including: a blank or NULL condition, "Password," "Admin," "Administrator," "sa," "sysadmin."

IMPORTANT A strong password cannot be the following terms associated with the installation computer: the name of the user currently logged onto the machine; the machine name.
A strong password must be more than 8 characters in length and satisfy at least three of the following four criteria: It must contain at least one number and must contain at least one non-alphanumeric character (e.g., #, %, or ^).

9. Change the installer suggested *hostname* to a DNS reachable hostname, or the IP address of the server.

NOTE The installer suggests the *hostname* of the computer as the default DNS reachable hostname. This value populates into sample data in the database and other sample files by the installer. This is used as part of URLs delivered to the CA50 to be used to access different services on the server. The default value that the installer presents should be changed if a CA50 would not be able to resolve the address via DNS. This can also be the static IP address of the server to avoid or work around any DNS resolution issues.

10. Click **Next** to continue. The *UCA Client Language* window displays.

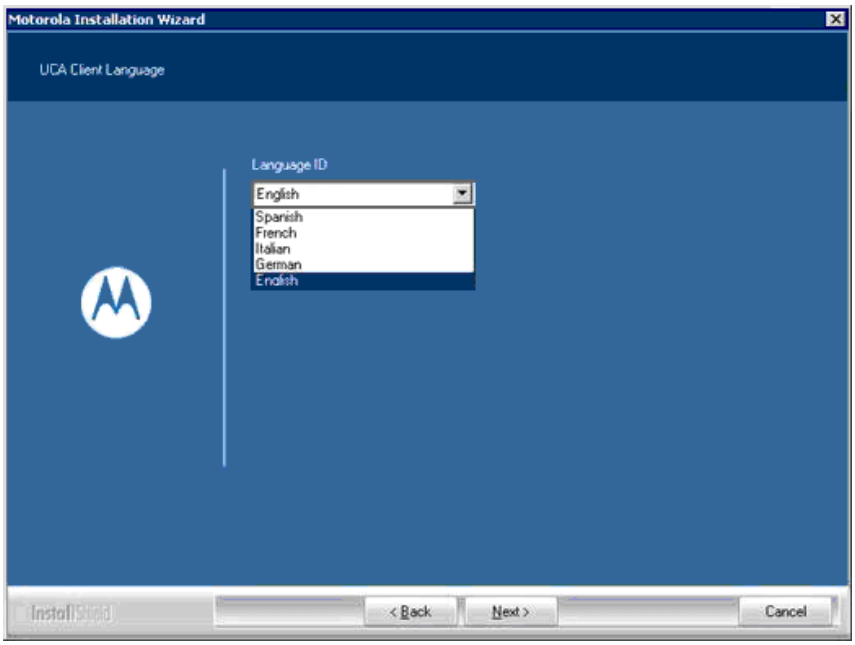


Figure 6-6 *UCA Client Language Window*

11. Select a language from the Language ID drop-down list. This is the language to display on the CA50 device at login. Click **Next** to continue. The *UCA Client Language* window displays.

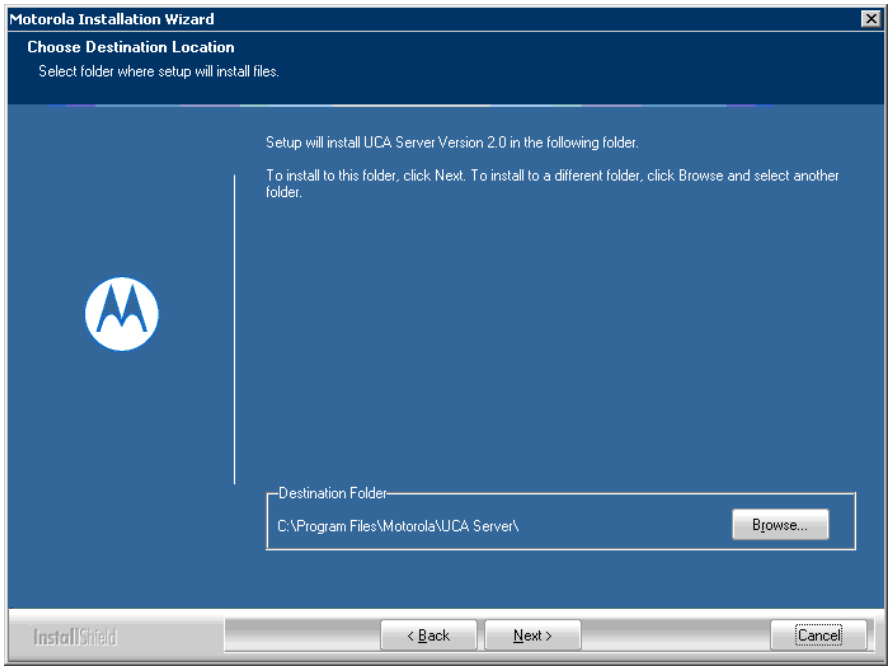

Motorola Installation Wizard
Choose Destination Location
Select folder where setup will install files.
Setup will install UCA Server Version 2.0 in the following folder.
To install to this folder, click Next. To install to a different folder, click Browse and select another folder.
Destination Folder
C:\Program Files\Motorola\UCA Server\
Browse...
InstallShield
< Back
Next >
Cancel

Figure 6-7 *Installation Destination Window*

12. Choose the destination directory in which to install the UCA server software. The default directory is C:\Program Files\Motorola\UCA Server.

13. Click **Next** to continue. The *Start Copying Files* window displays.

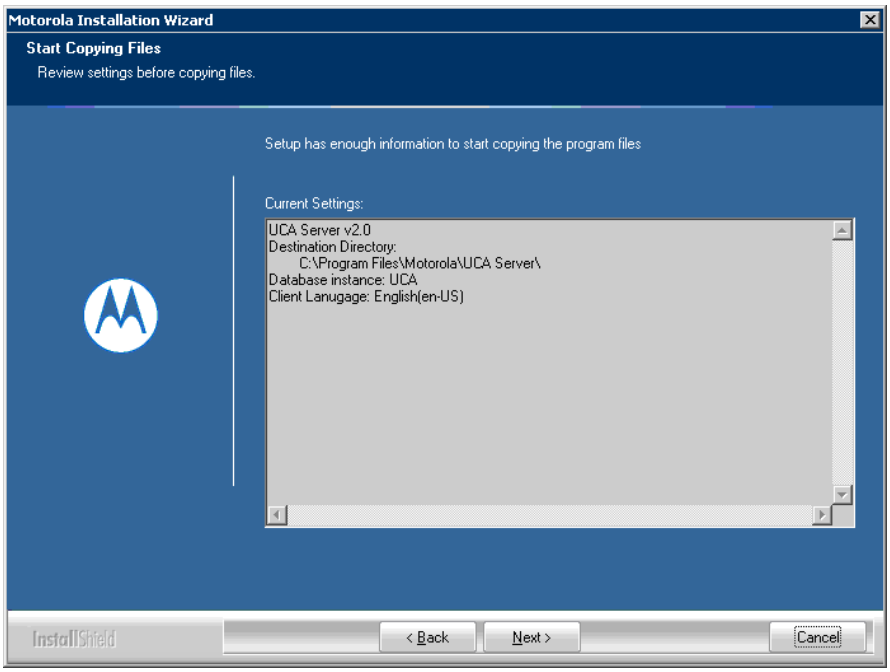


Figure 6-8 *Start Copying Files Window*

NOTE Ensure that the settings match the summary and client. The installation process may take some time to complete. Command windows flashing is typical during the installation process.

14. Click **Back** to change settings, or click **Next** to proceed with the installation.
The InstallShield begins to extract files and the install runs without further intervention.

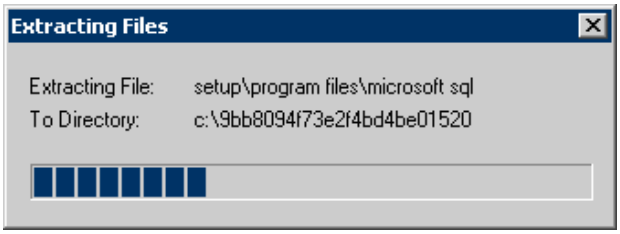


Figure 6-9 *Extracting Files*

IMPORTANT This installation requires Internet Information Services (IIS). If IIS is not installed on the host machine the installation aborts.

15. The SQL Server 2005 Express is installed. *Figure 6-10* to *Figure 6-16* display in sequence as the install proceeds.

IMPORTANT The installation takes approximately 15 to 20 minutes, depending on whether or not .NET Framework is already loaded, and the computer processor speed. During the installation process, the Installshield progress bar may not always appear to be moving. This is normal behavior and does not mean the installation process stopped.

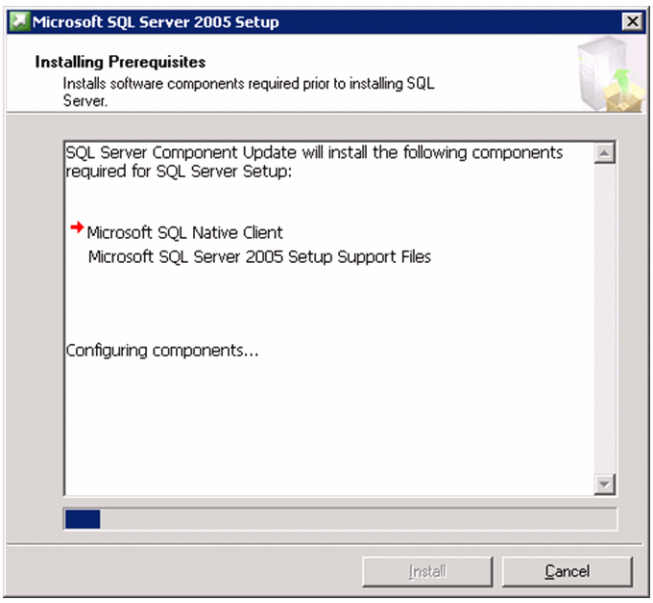

Microsoft SQL Server 2005 Setup
Installing Prerequisites
Installs software components required prior to installing SQL Server.
SQL Server Component Update will install the following components required for SQL Server Setup:
Microsoft SQL Native Client
Microsoft SQL Server 2005 Setup Support Files
Configuring components...
Install
Cancel

Figure 6-10 *Installing Prerequisites*

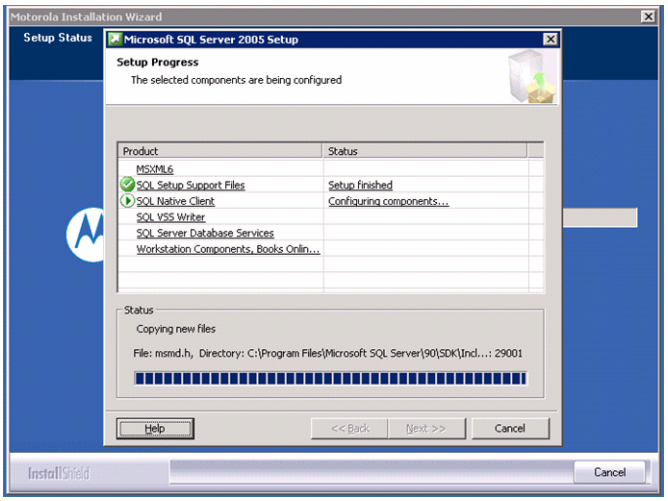


Figure 6-11 *SQL Server Setup*

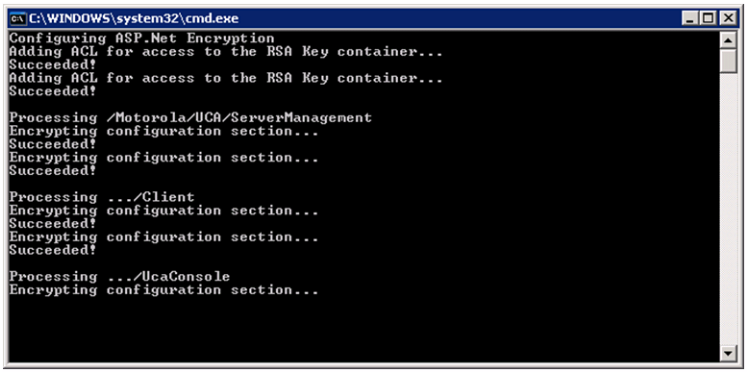


Figure 6-12 *Adding ACL*

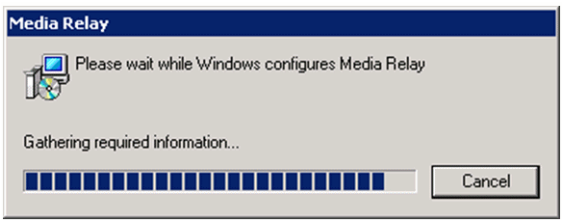


Figure 6-13 *Media Relay*

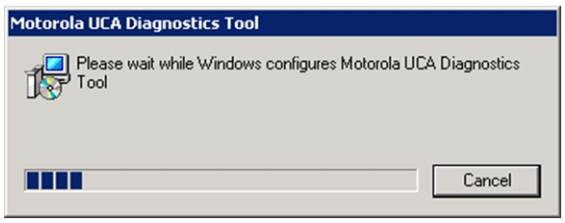


Figure 6-14 *UCA Diagnostics*

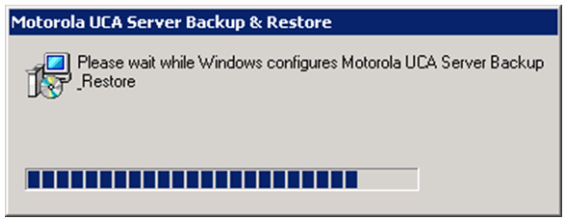


Figure 6-15 *UCA Server Backup & Restore*

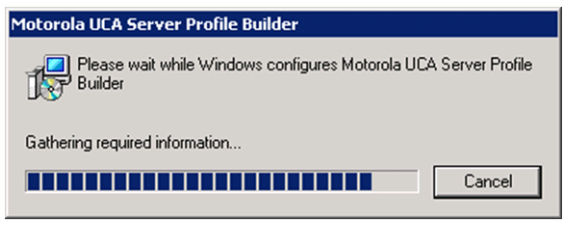

Motorola UCA Server Profile Builder
Please wait while Windows configures Motorola UCA Server Profile Builder
Gathering required information...
Cancel

Figure 6-16 *UCA Server Profile Builder*

16. Click **Yes** to start the console, or click **No**.

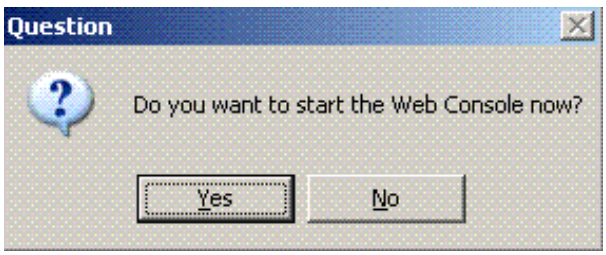


Figure 6-17 *Start Web Console*

17. If **Yes** is clicked to start the Web Console now the following Web Console window displays. See *Login to the UCA Server System on page 9-3* for login information.

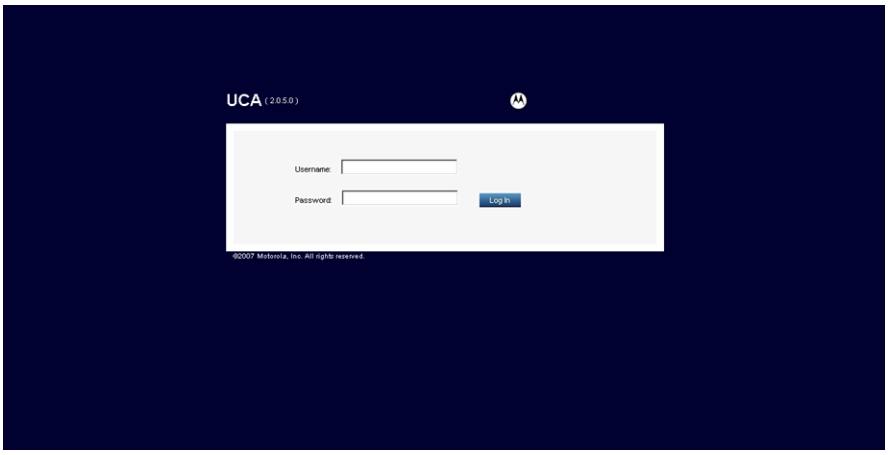


Figure 6-18 *WebConsole GUI Login*

IMPORTANT At the initial login to the Web Console (for details see *page 9-3*), it is required to use the default Web Console administrator *Username* and *Password*:

- *Factory Default Web Console Username*: *admin*
- *Factory Default Web Console Password*: *admin*

It is strongly suggested that the default administrator *Username* and *Password* be changed after the initial login. Admin Password can be changed on the UCA server console from System Management > Console Users > Console User Detail > Edit Console User Passwords.

Note that the *Username* and *Password* entered during the UCA server software installation process, (*step on page 6-13*) are for SQL database access and should not be mistaken with the default Web Console administrator Username and Password shown above.

18. The final dialog of the installation displays.

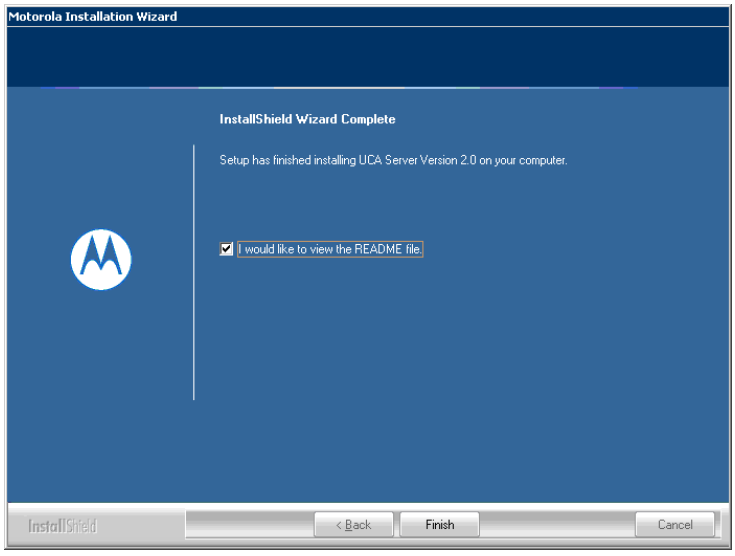


Figure 6-19 *InstallShield Wizard Complete*

19. To view the *Readme* document, leave the box checked and click **Finish**. To end the installation without viewing the *Readme* document, uncheck the box and click **Finish**.

20. After installing and launching the UCA server software, it is strongly recommended to validate that the server was successfully installed and is performing properly. For details see *Validate Successful Server Installation and Proper Performance on page 6-45*.

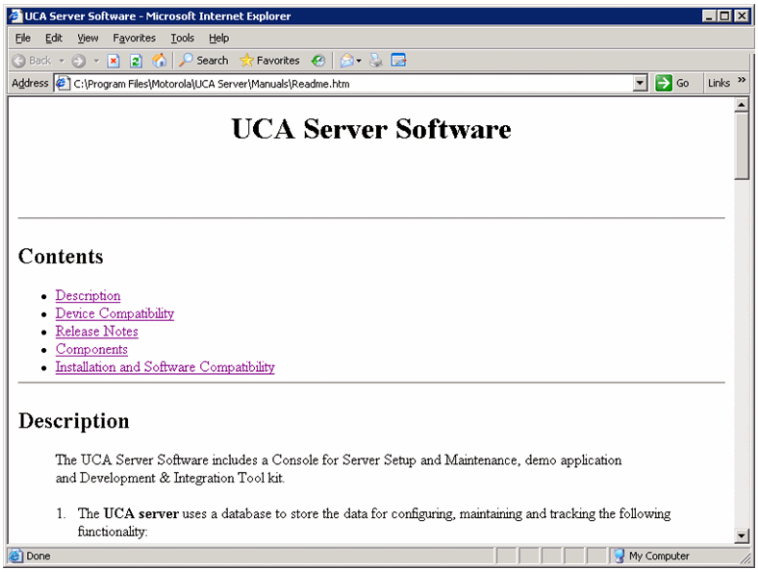

UCA Server Software - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Address C:\Program Files\Motorola\UCA Server\Manuals\Readme.htm
UCA Server Software
Contents
Description
Device Compatibility
Release Notes
Components
Installation and Software Compatibility
Description
The UCA Server Software includes a Console for Server Setup and Maintenance, demo application and Development & Integration Tool kit.
1. The UCA server uses a database to store the data for configuring, maintaining and tracking the following functionality:
Done
My Computer

Figure 6-20 *UCA Server Software Readme*

Installing Demo Version

To install the Demo version of the UCA server software:

1. Start the *UCA Server Software from the CA50* CD.

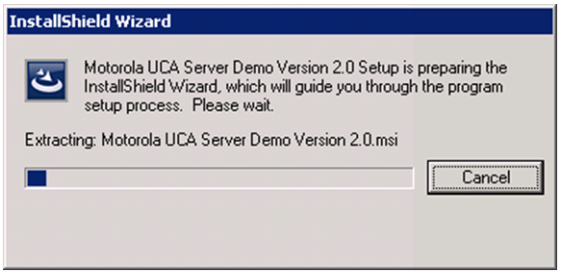

InstallShield Wizard
Motorola UCA Server Demo Version 2.0 Setup is preparing the InstallShield Wizard, which will guide you through the program setup process. Please wait.
Extracting: Motorola UCA Server Demo Version 2.0.msi
Cancel

Figure 6-21 *UCA Server Demo*

2. From the CD main menu, double-click the *UCA_Server_2.0.5_demo.exe*. The *Welcome* window displays.

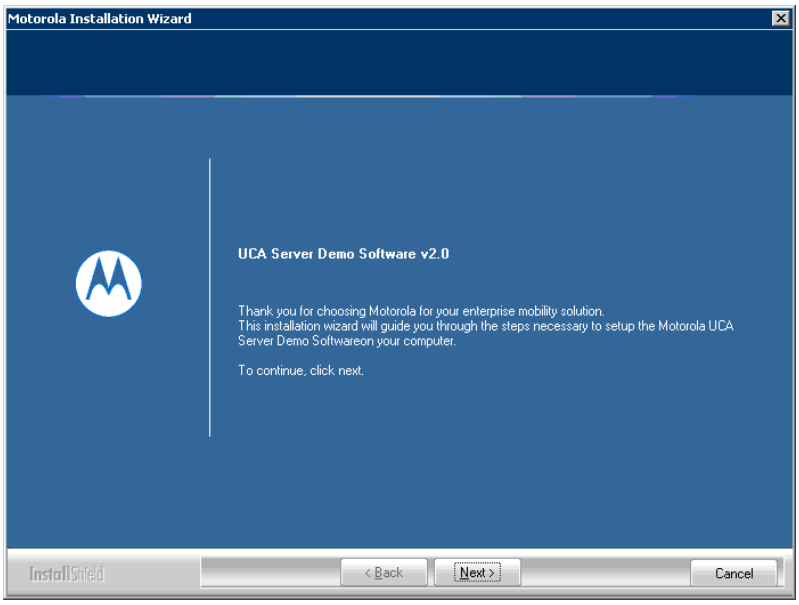

Motorola Installation Wizard
UCA Server Demo Software v2.0
Thank you for choosing Motorola for your enterprise mobility solution.
This installation wizard will guide you through the steps necessary to setup the Motorola UCA Server Demo Softwareon your computer.
To continue, click next.
InstallShield
< Back
Next >
Cancel

Figure 6-22 *Welcome*

3. Click **Next** to continue. The *Customer Information window* displays.

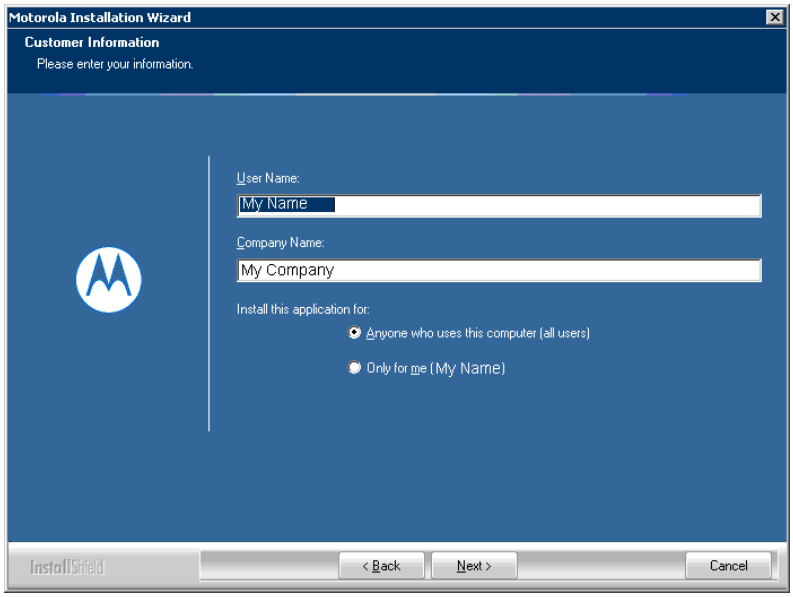


Figure 6-23 *Customer Information Window*

4. Perform *step 7 on page 6-13* through *step 17 on page 6-18* to complete the installation.

5. Enter a strong password[1] to use for SQL server authentication.

NOTE The password entered in this field is the system administrator (sa) password. The sa account in SQL Server 2005 has full access and read/write permissions to all DB instances. It is used during the installation process and for future upgrades. This password should be recorded and retained in a safe location by the integrator or customer.

IMPORTANT Microsoft Strong Password Guidelines for SQL Server 2005 authentication:
Strong passwords are not readily guessed by a person, and are not easily hacked using a computer program. Strong passwords cannot use prohibited conditions or terms, including: a blank or NULL condition, "Password," "Admin," "Administrator," "sa," "sysadmin."

IMPORTANT A strong password cannot be the following terms associated with the installation computer: the name of the user currently logged onto the machine; the machine name.
A strong password must be more than 8 characters in length and satisfy at least three of the following four criteria: it must contain at least one number and must contain at least one non-alphanumeric character (e.g., #, %, or ^).

6. Change the installer suggested *hostname* to a DNS reachable hostname, or the IP address of the server.

NOTE The installer suggests the *hostname* of the computer as the default DNS reachable hostname. This value populates into sample data in the database and other sample files by the installer. This is used as part of URLs delivered to the CA50 to be used to access different services on the server. The default value that the installer presents should be changed if a CA50 would not be able to resolve the address via DNS. This can also be the static IP address of the server to avoid or work around any DNS resolution issues.

7. Enter a strong password for SQL database access.

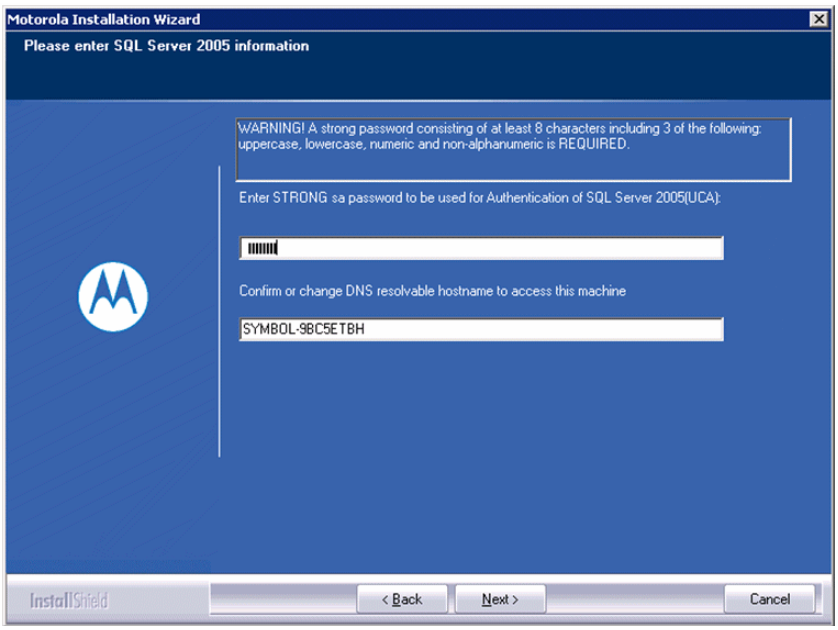

Motorola Installation Wizard
Please enter SQL Server 2005 information
WARNING! A strong password consisting of at least 8 characters including 3 of the following: uppercase, lowercase, numeric and non-alphanumeric is REQUIRED.
Enter STRONG sa password to be used for Authentication of SQL Server 2005(UCA):
Confirm or change DNS resolvable hostname to access this machine
SYMBOL-9BC5ETBH
InstallShield
< Back
Next >
Cancel

Figure 6-24 *SQL Server 2005 Information*

8. Enter the user name (Login ID) and the password for SQL database access.

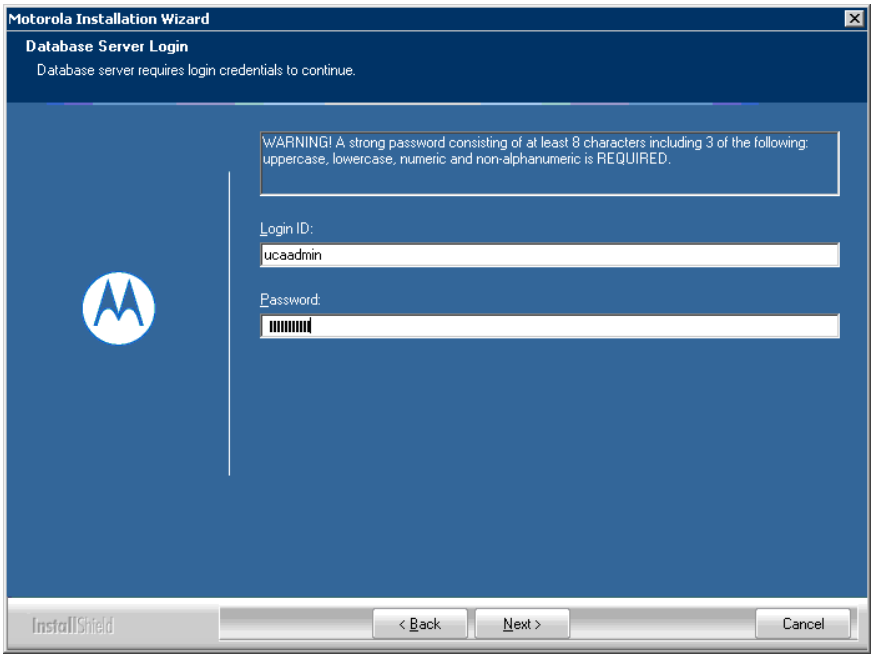

Motorola Installation Wizard
Database Server Login
Database server requires login credentials to continue.
WARNING! A strong password consisting of at least 8 characters including 3 of the following: uppercase, lowercase, numeric and non-alphanumeric is REQUIRED.
Login ID:
ucaadmin
Password:
InstallShield
< Back
Next >
Cancel

Figure 6-25 *SQL Database Server Login*

IMPORTANT [1]Microsoft Strong Password Guidelines for SQL Server 2005 authentication:
Strong passwords are not readily guessed by a person, and are not easily hacked using a computer program. Strong passwords cannot use prohibited conditions or terms, including: a blank or NULL condition, "Password," "Admin," "Administrator," "sa," "sysadmin." Password criteria is a minimum of 8 ASCII characters.

9. Click **Next** to continue. The *UCA Client Language* window displays.

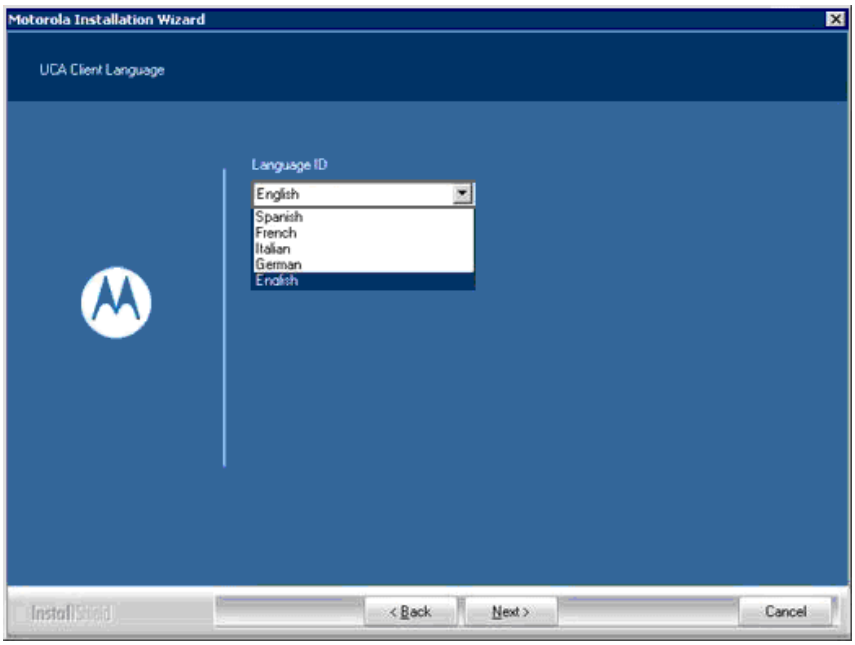


Figure 6-26 *UCA Client Language Window*

10. Select a language from the Language ID drop-down list. This is the language to display on the CA50 device at login. Click **Next** to continue. The *UCA Client Language* window displays.

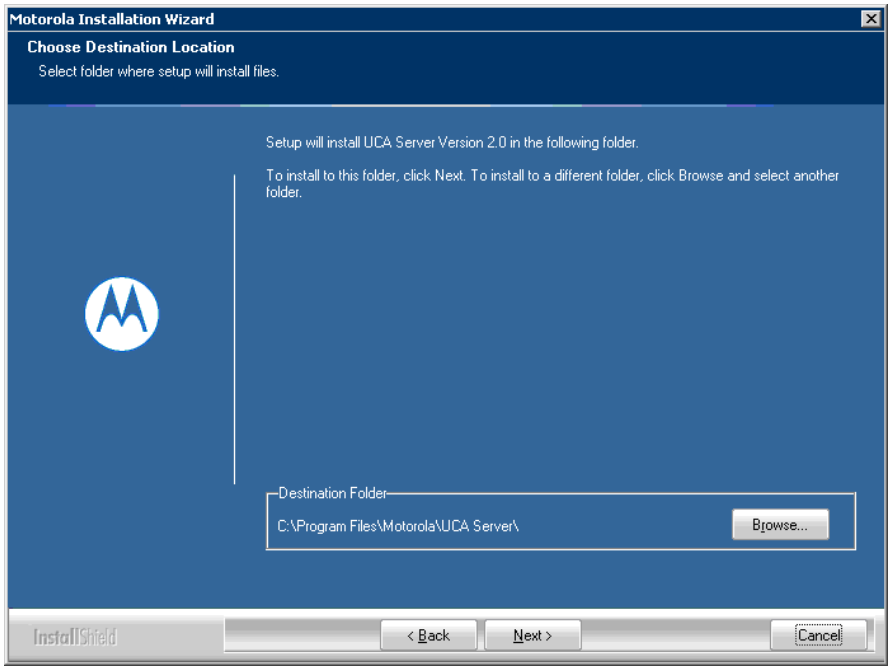

Motorola Installation Wizard
Choose Destination Location
Select folder where setup will install files.
Setup will install UCA Server Version 2.0 in the following folder.
To install to this folder, click Next. To install to a different folder, click Browse and select another folder.
Destination Folder
C:\Program Files\Motorola\UCA Server\
Browse...
InstallShield
< Back
Next >
Cancel

Figure 6-27 *Installation Destination Window*

11. Choose the destination directory in which to install the UCA server software. The default directory is C:\Program Files\Motorola\UCA Server.

12. Click **Next** to continue. The *Start Copying Files* window displays.

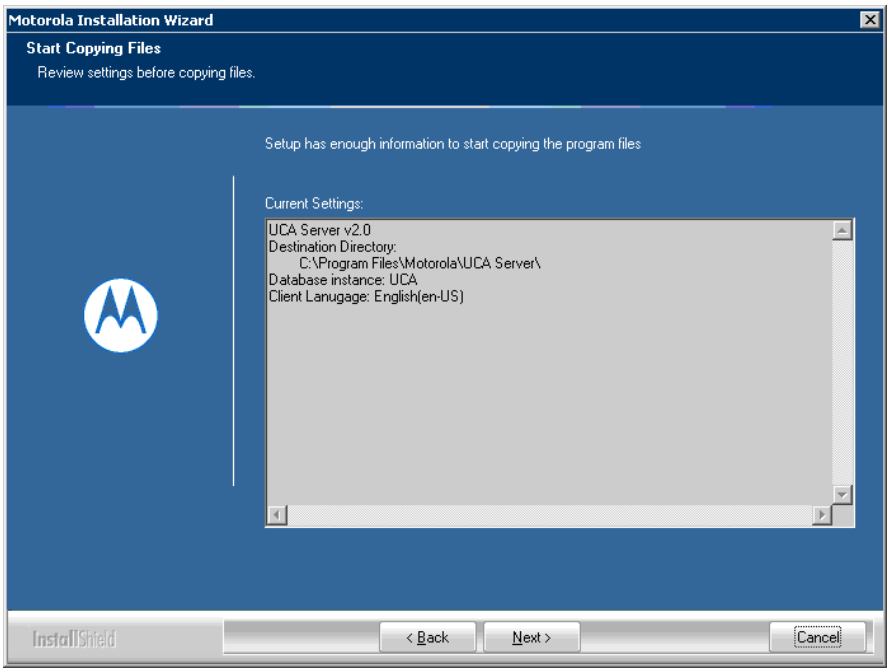


Figure 6-28 *Start Copying Files Window*

NOTE Ensure that the settings match the summary and client. The installation process may take some time to complete. Command windows flashing is typical during the installation process.

13. Click **Back** to change settings, or click **Next** to proceed with the installation.
The InstallShield begins to extract files and the install runs without further intervention.

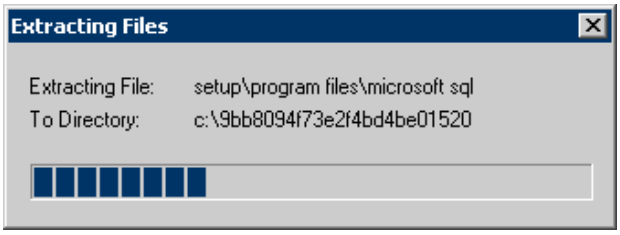


Figure 6-29 *Extracting Files*

IMPORTANT This installation requires Internet Information Services (IIS). If IIS is not installed on the host machine the installation aborts.

14. The SQL Server 2005 Express is installed. *Figure 6-30* to *Figure 6-39* display in sequence as the install proceeds.

IMPORTANT The installation takes approximately 15 to 20 minutes, depending on whether or not .NET Framework is already loaded, and the computer processor speed. During the installation process, the Installshield progress bar may not always appear to be moving. This is normal behavior and does not mean the installation process stopped.

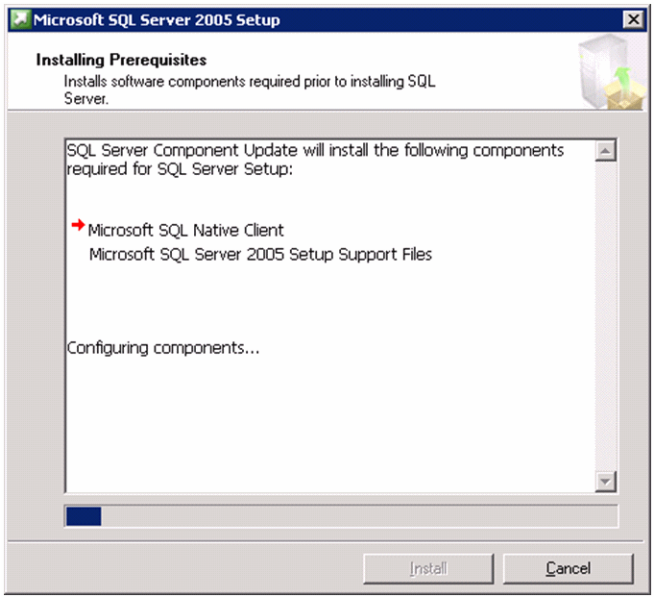

Microsoft SQL Server 2005 Setup
Installing Prerequisites
Installs software components required prior to installing SQL Server.
SQL Server Component Update will install the following components required for SQL Server Setup:
Microsoft SQL Native Client
Microsoft SQL Server 2005 Setup Support Files
Configuring components...
Install
Cancel

Figure 6-30 *Installing Prerequisites*

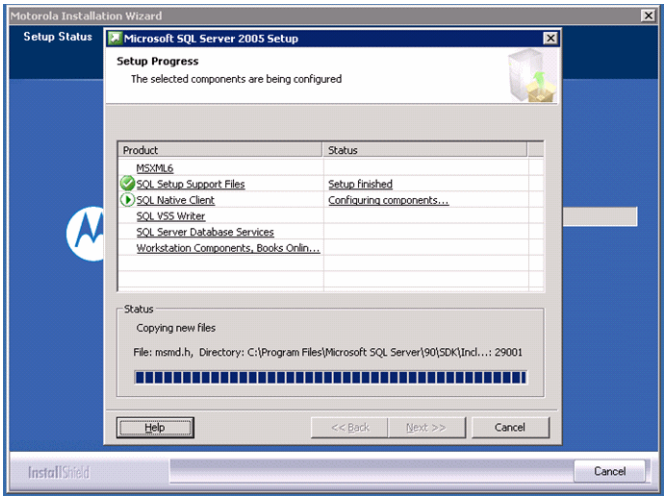


Figure 6-31 *SQL Server Setup*

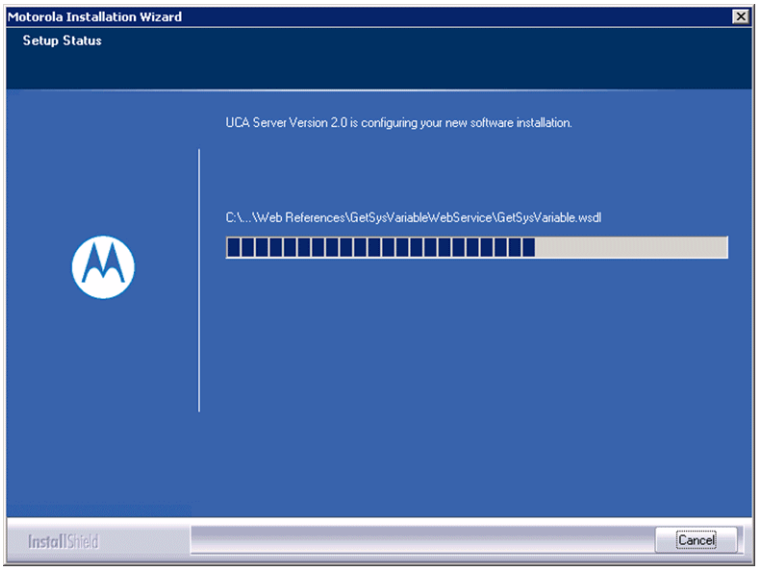


Figure 6-32 *SQL Server Setup Status*

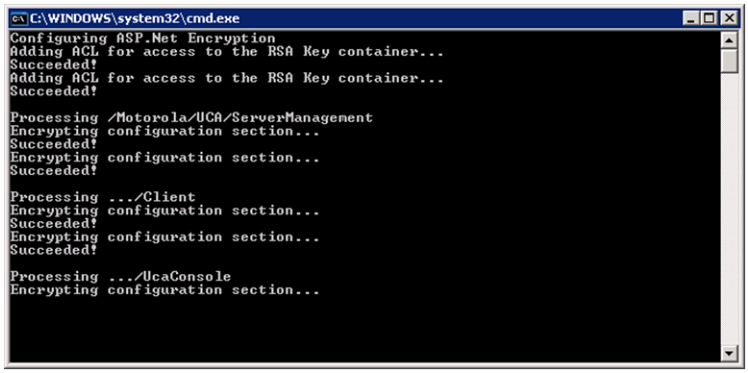


Figure 6-33 *Adding ACL*

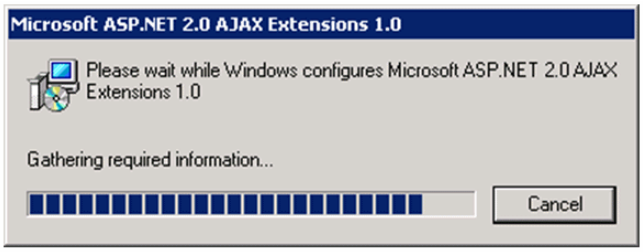

Microsoft ASP.NET 2.0 AJAX Extensions 1.0
Please wait while Windows configures Microsoft ASP.NET 2.0 AJAX Extensions 1.0
Gathering required information...
Cancel

Figure 6-34 *ASP.NET 2.0 Ajax Extensions 1.0*

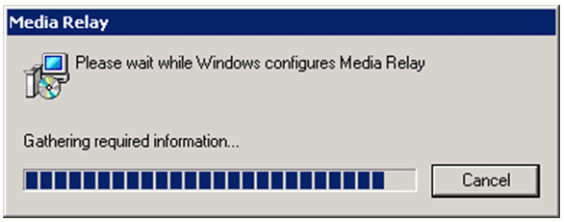


Figure 6-35 *Media Relay*

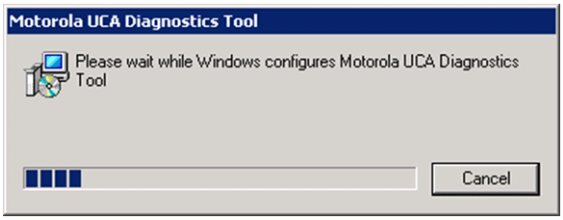


Figure 6-36 *UCA Diagnostics*

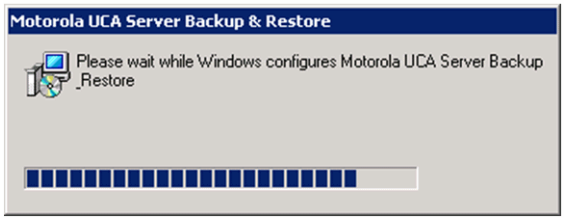


Figure 6-37 *UCA Server Backup & Restore*

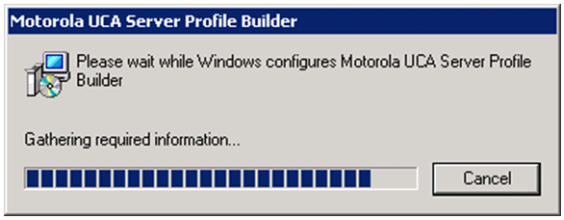

Motorola UCA Server Profile Builder
Please wait while Windows configures Motorola UCA Server Profile Builder
Gathering required information...
Cancel

Figure 6-38 *UCA Server Profile Builder*

15. Click **Yes** to start the console, or click **No**.

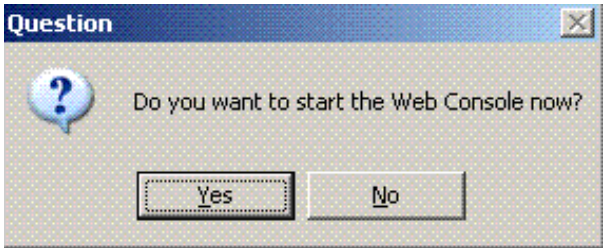


Figure 6-39 *Start Web Console*

16. The final dialog of the installation displays.

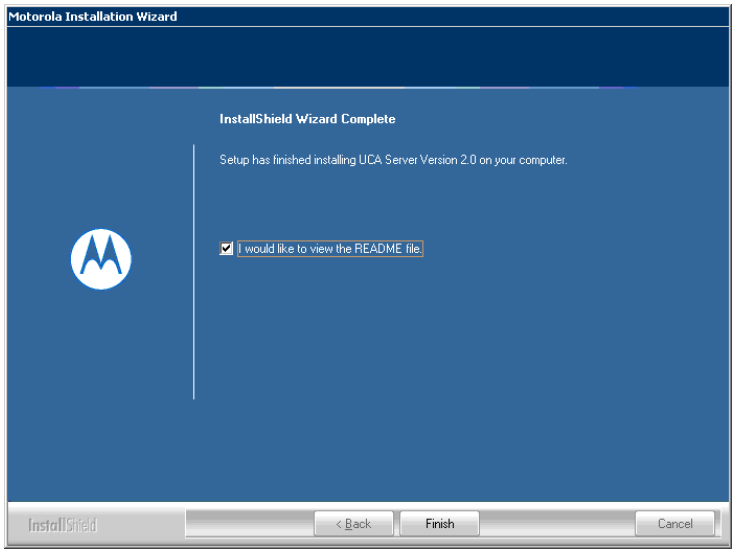


Figure 6-40 *InstallShield Wizard Complete*

17. To view the *Readme* document, leave the box checked and click **Finish**. To end the installation without viewing the *Readme* document, uncheck the box and click **Finish**.

18. After installing and launching the UCA server software, it is strongly recommended to validate that the server was successfully installed and is performing properly. For details see *Validate Successful Server Installation and Proper Performance on page 6-45*..

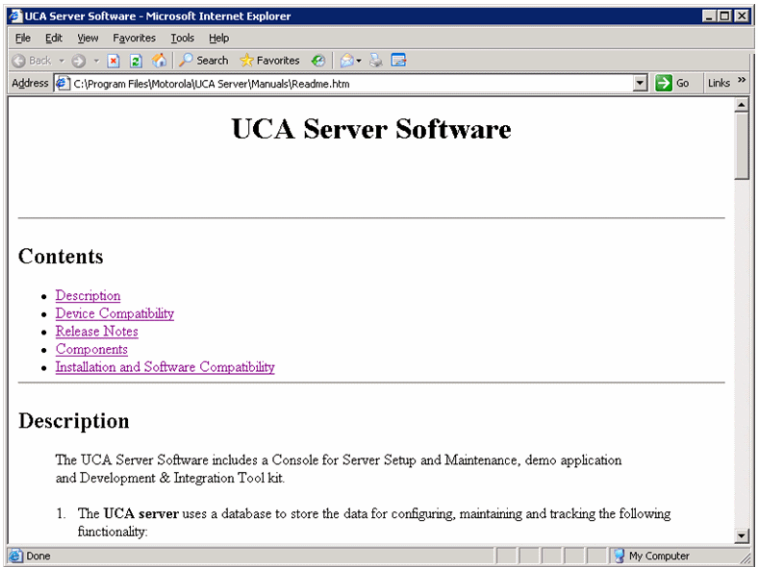

UCA Server Software - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Address C:\Program Files\Motorola\UCA Server\Manuals\Readme.htm
UCA Server Software
Contents
Description
Device Compatibility
Release Notes
Components
Installation and Software Compatibility
Description
The UCA Server Software includes a Console for Server Setup and Maintenance, demo application and Development & Integration Tool kit.
1. The UCA server uses a database to store the data for configuring, maintaining and tracking the following functionality:
Done
My Computer

Figure 6-41 *UCA Server Software Readme*

19. If **Yes** is clicked to start the Web Console now the following Web Console window displays. See *Login to the UCA Server System on page 9-3* for login information.

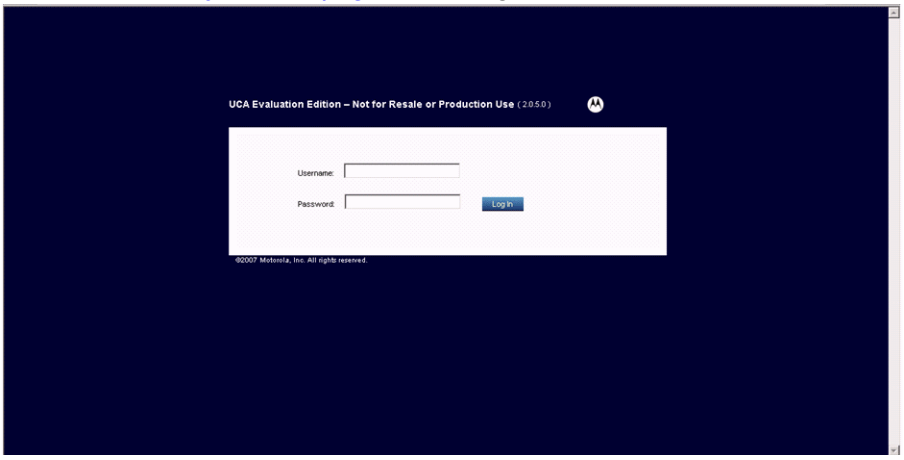

UCA Evaluation Edition – Not for Resale or Production Use (2.0.5.0)
Username:
Password:
Log In
©2007 Motorola, Inc. All rights reserved.

Figure 6-42 *UCA Login*

IMPORTANT At the initial login to the Web Console (for details see *page 9-3*), it is required to use the default Web Console administrator *Username* and *Password*:

- *Factory Default Web Console Username*: *admin*
- *Factory Default Web Console Password*: *admin*

It is strongly suggested that the default administrator *Username* and *Password* be changed after the initial login. Admin Password can be changed on the UCA server console from System Management > Console Users > Console User Detail > Edit Console User Passwords.

Note that the *Username* and *Password* entered during the UCA server software installation process, (*step on page 6-13*) are for SQL database access and should not be mistaken with the default Web Console administrator Username and Password shown above.

Upgrading the Demo Version to Production

To upgrade the Demo version of the UCA server software to the Production version:

1. Start the *UCA Server Software for the CA50* CD.
2. From the CD main menu, double-click the “Motorola_UCA_Server_2.0.5.exe”. The Welcome window displays.

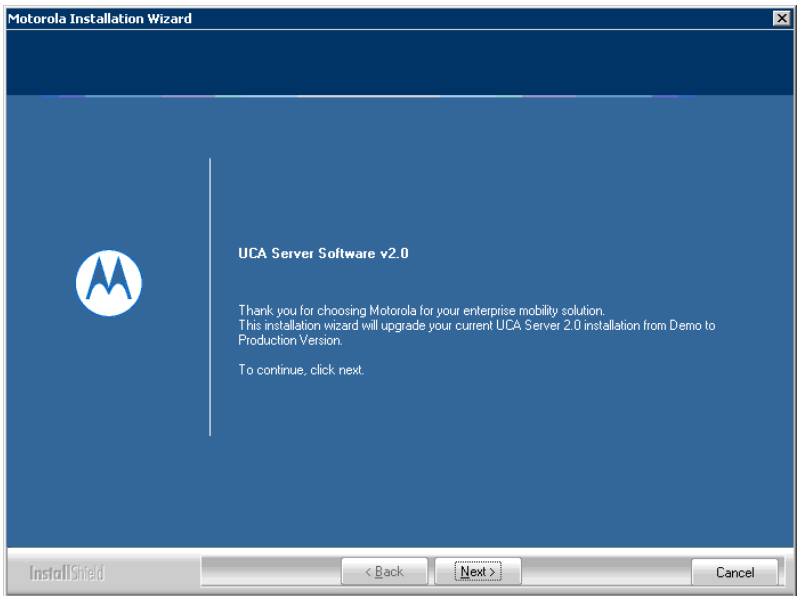

Motorola Installation Wizard
UCA Server Software v2.0
Thank you for choosing Motorola for your enterprise mobility solution.
This installation wizard will upgrade your current UCA Server 2.0 installation from Demo to Production Version.
To continue, click next.
InstallShield
< Back
Next >
Cancel

Figure 6-43 *Welcome*

3. Click **Next** on the *Welcome* window.
   The *License Agreement* displays.

Motorola Installation Wizard
License Agreement
Please read the following license agreement carefully.
MOTOROLA, INC.
END USER LICENSE AGREEMENT
Motorola is willing to license its Software and the accompanying documentation to you only on the condition that you accept all the terms in this License Agreement ("Agreement").
IMPORTANT: READ THE FOLLOWING TERMS AND CONDITIONS BEFORE USING THE ACCOMPANYING PRODUCTS.
BY CLICKING ON THE "ACCEPT" BUTTON BELOW, YOU ACKNOWLEDGE THAT YOU HAVE READ THIS AGREEMENT, UNDERSTAND IT AND AGREE TO BE BOUND BY THE TERMS OF THIS AGREEMENT. IF YOU DO NOT AGREE TO THE TERMS OF THIS AGREEMENT, MOTOROLA IS NOT WILLING TO LICENSE THE PRODUCTS TO YOU. YOU SHOULD CLICK ON THE "DO NOT ACCEPT" BUTTON TO DISCONTINUE THE SOFTWARE INSTALLATION PROCESS. IF YOU DO NOT AGREE TO THESE TERMS, YOU MAY, WITHIN 15 DAYS, RETURN THIS ENTIRE PRODUCT TO THE LOCATION WHERE YOU ACQUIRED IT OR PROVIDE WRITTEN VERIFICATION OF DELETION OF ALL COPIES OF THE ENTIRE PRODUCT IF YOU HAVE NOT PHYSICALLY RECEIVED A PRODUCT FOR A FULL REFUND.
1. Definitions.
I accept the terms of the license agreement
Print
I do not accept the terms of the license agreement
InstallShield
< Back
Next >
Cancel

Figure 6-44 *License Agreement*

4. Read the license agreement and click **Yes** to accept and continue the installation.
   The *Customer Information* window displays with the current user name and company. Change if necessary.

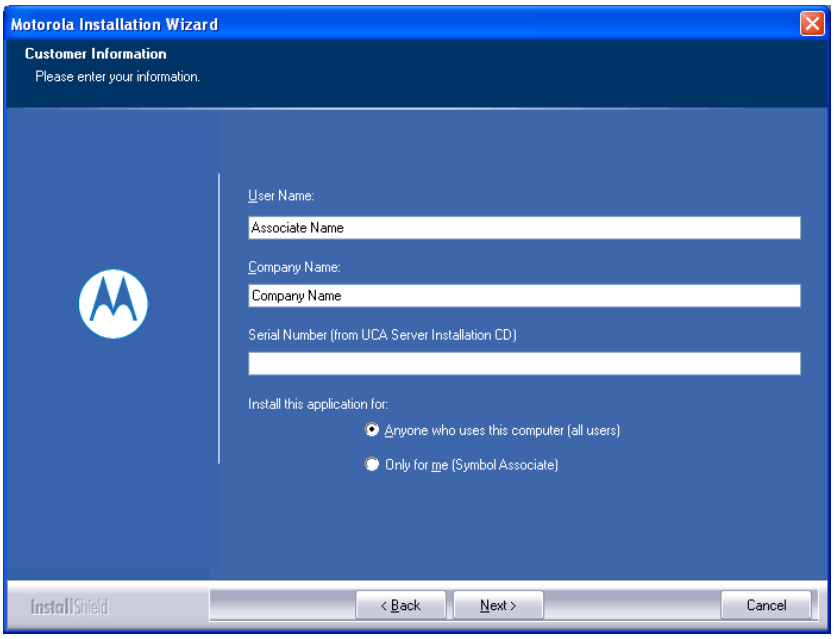


Figure 6-45 *Customer Information*

5. Enter a valid serial number and click **Next** to begin the upgrade process.

NOTE This process takes some time to complete.

Upgrading Software Revisions from Revision A to Revision B

IMPORTANT Before starting an upgrade, back up the older version of the UCA database (e.g., version 1.4.x.x). The Backup utility creates a file called UCA_Backup_DDMMYYYYHHMMSS.zip. See *UCA Backup & Restore on page H-1* for detailed back up information.

To upgrade to a new revision of the UCA server software:

1. Back up the existing version of the UCA database using the backup and restore utility. This utility creates a file called UCA_Backup_DDMMYYYYHHMMSS.zip which contains UCA database and user profiles backup.
2. Extract the zipped contents of the backed up UCA database to any local folder on the PC hard drive (e.g., C:\Temp).
3. Uninstall the old version of UCA server software from the computer (see *Uninstalling UCA Server Software on page 6-42*).
4. Start the *UCA Server Software for the CA50* CD.

5. From the CD main menu, double-click the *"Motorola_UCA_Server_2.0.5_upgrade.exe"* to begin the upgrade.

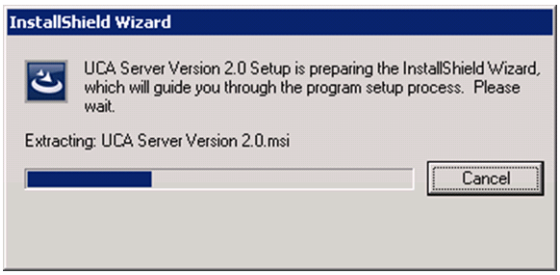

InstallShield Wizard
UCA Server Version 2.0 Setup is preparing the InstallShield Wizard, which will guide you through the program setup process. Please wait.
Extracting: UCA Server Version 2.0.msi
Cancel

Figure 6-46 *Install UCA Server Software*

6. This may take several minutes, before the *Welcome* window displays.

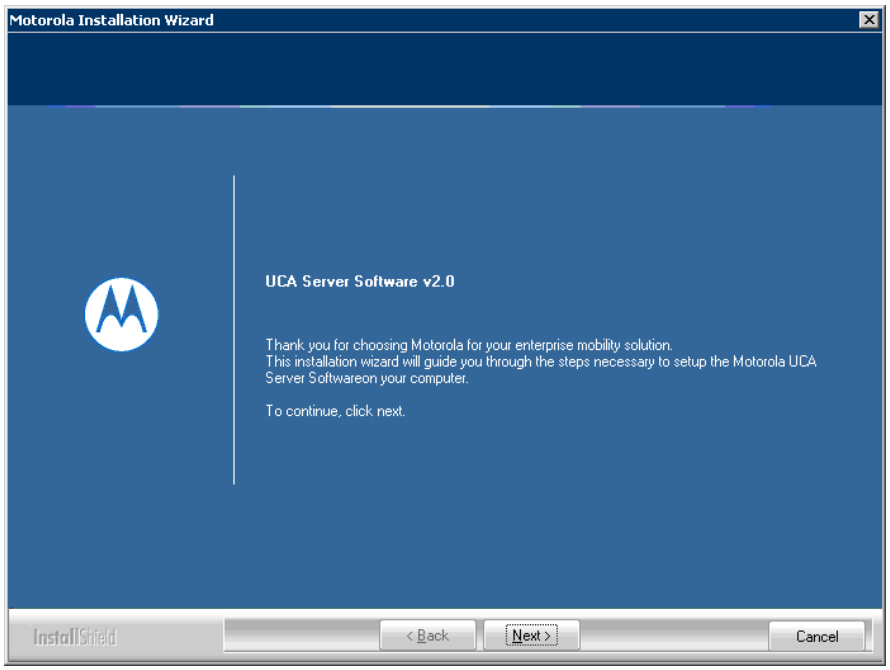

Motorola Installation Wizard
UCA Server Software v2.0
Thank you for choosing Motorola for your enterprise mobility solution.
This installation wizard will guide you through the steps necessary to setup the Motorola UCA Server Softwareon your computer.
To continue, click next.
InstallShield
< Back
Next >
Cancel

Figure 6-47 *Welcome*

7. Click **Next** on the *Welcome* window.
   The *License Agreement* displays.

Motorola Installation Wizard
License Agreement
Please read the following license agreement carefully.
MOTOROLA, INC.
END USER LICENSE AGREEMENT
Motorola is willing to license its Software and the accompanying documentation to you only on the condition that you accept all the terms in this License Agreement ("Agreement").
IMPORTANT: READ THE FOLLOWING TERMS AND CONDITIONS BEFORE USING THE ACCOMPANYING PRODUCTS.
BY CLICKING ON THE "ACCEPT" BUTTON BELOW, YOU ACKNOWLEDGE THAT YOU HAVE READ THIS AGREEMENT, UNDERSTAND IT AND AGREE TO BE BOUND BY THE TERMS OF THIS AGREEMENT. IF YOU DO NOT AGREE TO THE TERMS OF THIS AGREEMENT, MOTOROLA IS NOT WILLING TO LICENSE THE PRODUCTS TO YOU. YOU SHOULD CLICK ON THE "DO NOT ACCEPT" BUTTON TO DISCONTINUE THE SOFTWARE INSTALLATION PROCESS. IF YOU DO NOT AGREE TO THESE TERMS, YOU MAY, WITHIN 15 DAYS, RETURN THIS ENTIRE PRODUCT TO THE LOCATION WHERE YOU ACQUIRED IT OR PROVIDE WRITTEN VERIFICATION OF DELETION OF ALL COPIES OF THE ENTIRE PRODUCT IF YOU HAVE NOT PHYSICALLY RECEIVED A PRODUCT FOR A FULL REFUND.
1. Definitions.
I accept the terms of the license agreement
Print
I do not accept the terms of the license agreement
InstallShield
< Back
Next >
Cancel

Figure 6-48 *License Agreement*

8. Read the license agreement and click **Yes** to accept and continue the installation.
   The *Customer Information* window displays with the current user name and company. Change if necessary.

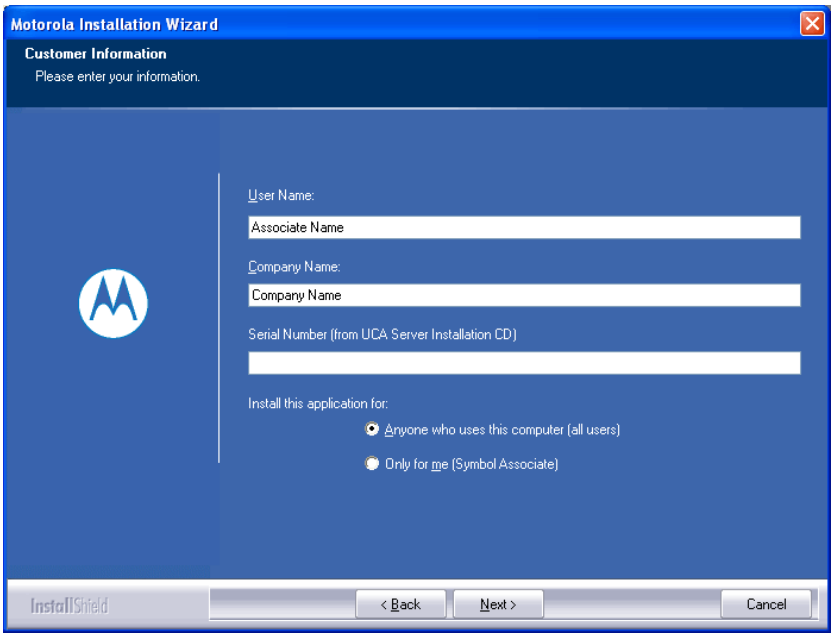


Figure 6-49 *Customer Information*

9. Enter the serial number that appears on the back of the UCA installation CD sleeve.

10. Click **Next**.
    The *SQL Server 2005 Information* window displays.

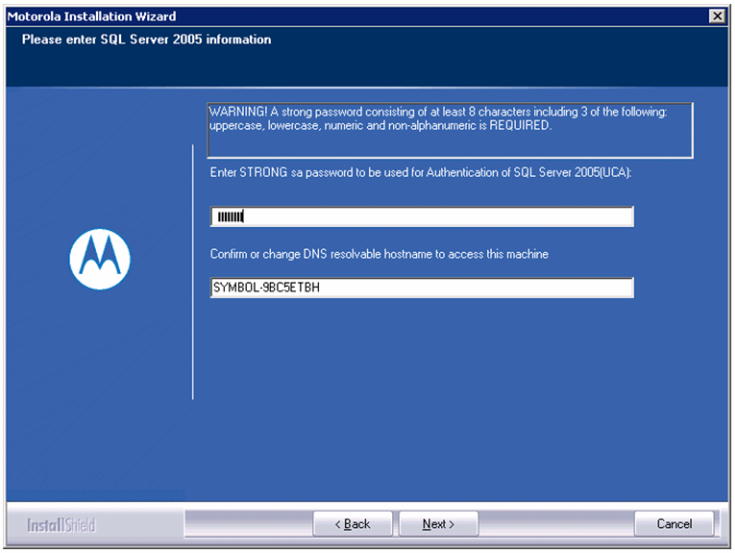

Motorola Installation Wizard
Please enter SQL Server 2005 information
WARNING! A strong password consisting of at least 8 characters including 3 of the following: uppercase, lowercase, numeric and non-alphanumeric is REQUIRED.
Enter STRONG sa password to be used for Authentication of SQL Server 2005(UCA):
Confirm or change DNS resolvable hostname to access this machine
SYMBOL-9BC5ETBH
InstallShield
< Back
Next >
Cancel

Figure 6-50 *SQL Server 2005 Information*

11. Enter a strong password[1] to use for SQL server authentication.

NOTE The password entered in this field is the system administrator (sa) password. The sa account in SQL Server 2005 has full access and read/write permissions to all DB instances. It is used during the installation process and for future upgrades. This password should be recorded and retained in a safe location by the integrator or customer.

IMPORTANT Microsoft Strong Password Guidelines for SQL Server 2005 authentication:
Strong passwords are not readily guessed by a person, and are not easily hacked using a computer program. Strong passwords cannot use prohibited conditions or terms, including: a blank or NULL condition, "Password," "Admin," "Administrator," "sa," "sysadmin."

IMPORTANT A strong password cannot be the following terms associated with the installation computer: the name of the user currently logged onto the machine; the machine name.
A strong password must be more than 8 characters in length and satisfy at least three of the following four criteria: It must contain at least one number and must contain at least one non-alphanumeric character (e.g., #, %, or ^).

12. Change the installer suggested *hostname* to a DNS reachable hostname, or the IP address of the server.

NOTE The installer suggests the *hostname* of the computer as the default DNS reachable hostname. This value populates into sample data in the database and other sample files by the installer. This is used as part of URLs delivered to the CA50 to be used to access different services on the server. The default value that the installer presents should be changed if a CA50 would not be able to resolve the address via DNS. This can also be the static IP address of the server to avoid or work around any DNS resolution issues.

13. Enter the user name (Login ID) and a strong password for SQL database access.

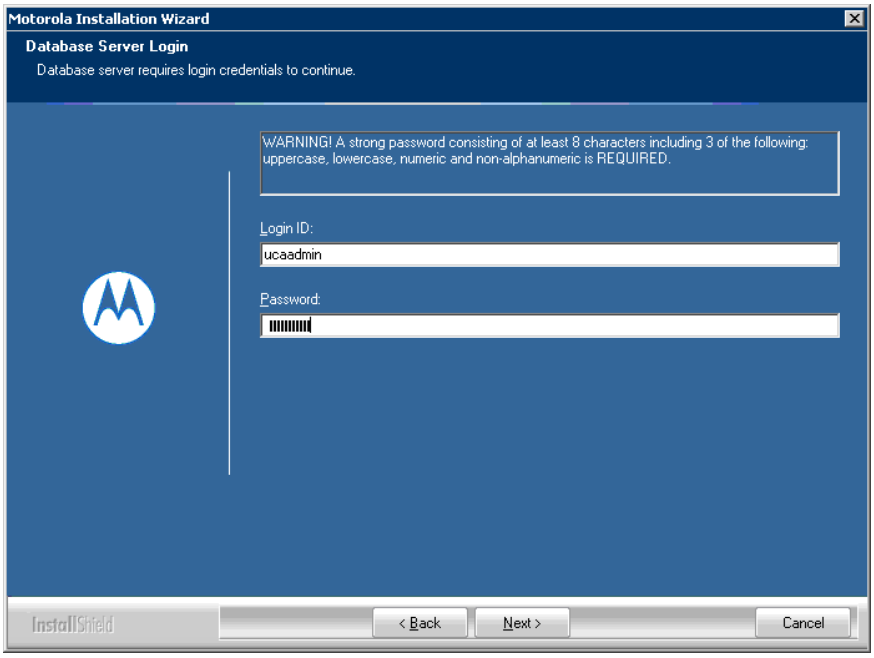

Motorola Installation Wizard
Database Server Login
Database server requires login credentials to continue.
WARNING! A strong password consisting of at least 8 characters including 3 of the following: uppercase, lowercase, numeric and non-alphanumeric is REQUIRED.
Login ID:
ucaadmin
Password:
InstallShield
< Back
Next >
Cancel

Figure 6-51 *SQL Database Server Login*

IMPORTANT [1]Microsoft Strong Password Guidelines for SQL Server 2005 authentication:
Strong passwords are not readily guessed by a person, and are not easily hacked using a computer program. Strong passwords cannot use prohibited conditions or terms, including: a blank or NULL condition, "Password," "Admin," "Administrator," "sa," "sysadmin." Password criteria is a minimum of 8 ASCII characters.

14. Enter a database user account details. Do not enter user names like "UCA", "SA".

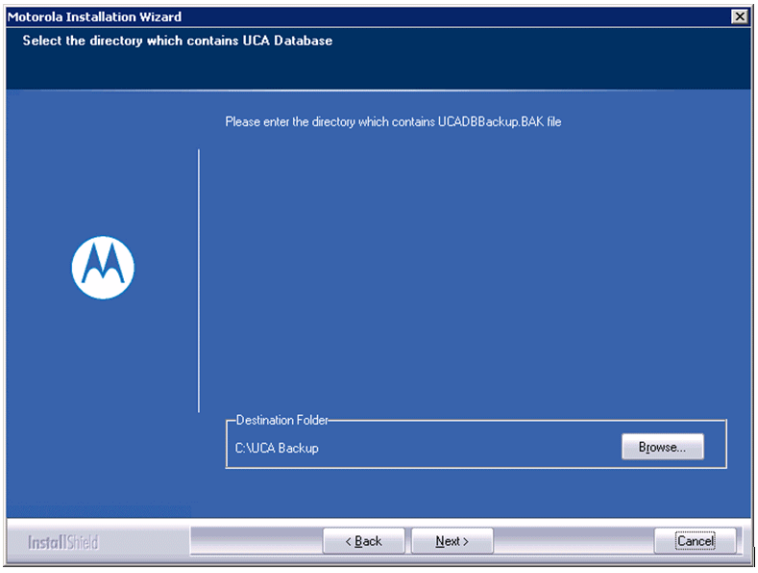


Figure 6-52 *Database User Account Details*

15. Browse to the folder to which backup zip file has been extracted.

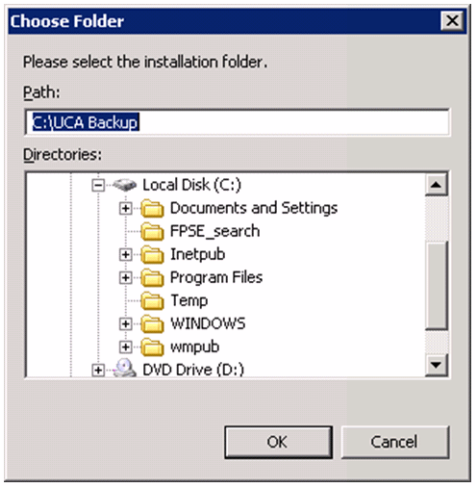


Figure 6-53 *Backup Folder*

16. Click **Next** to continue. The *UCA Client Language* window displays.

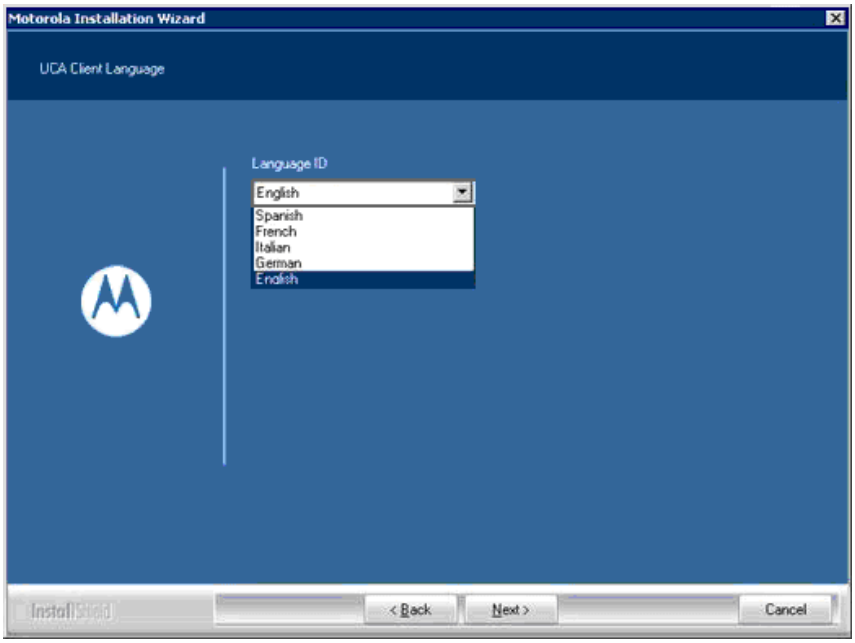


Figure 6-54 *UCA Client Language Window*

17. Select a language from the Language ID drop-down list. This is the language to display on the CA50 device at login. Click **Next** to continue. The *UCA Client Language* window displays.

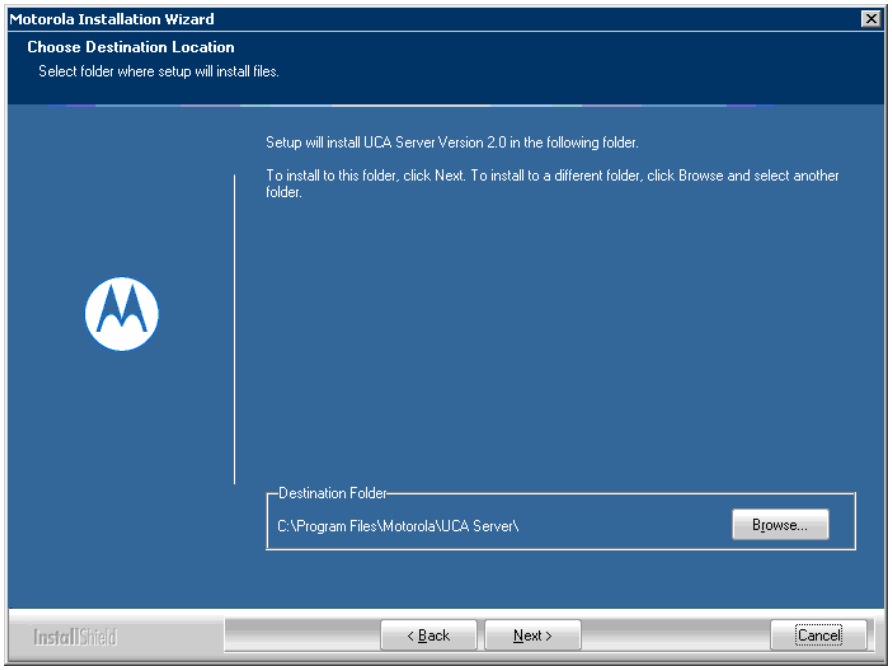

Motorola Installation Wizard
Choose Destination Location
Select folder where setup will install files.
Setup will install UCA Server Version 2.0 in the following folder.
To install to this folder, click Next. To install to a different folder, click Browse and select another folder.
Destination Folder
C:\Program Files\Motorola\UCA Server\
Browse...
InstallShield
< Back
Next >
Cancel

Figure 6-55 *Installation Destination Window*

18. Choose the destination directory in which to install the UCA server software. The default directory is C:\Program Files\Motorola\UCA Server.

19. Click **Back** to change settings, or click **Next** to proceed with the installation.
The InstallShield begins to extract files and the install runs without further intervention.

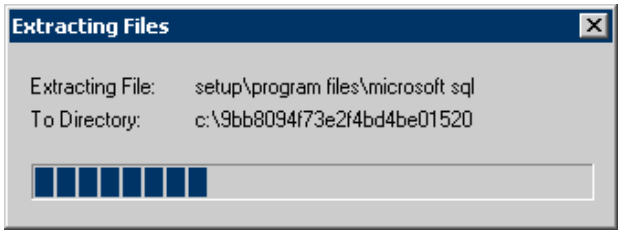


Figure 6-56 *Extracting Files*

IMPORTANT This installation requires Internet Information Services (IIS). If IIS is not installed on the host machine the installation aborts.

20. The SQL Server 2005 Express is installed. *Figure 6-57* to *Figure 6-64* display in sequence as the install proceeds.

IMPORTANT The installation takes approximately 15 to 20 minutes, depending on whether or not .NET Framework is already loaded, and the computer processor speed. During the installation process, the Installshield progress bar may not always appear to be moving. This is normal behavior and does not mean the installation process stopped.

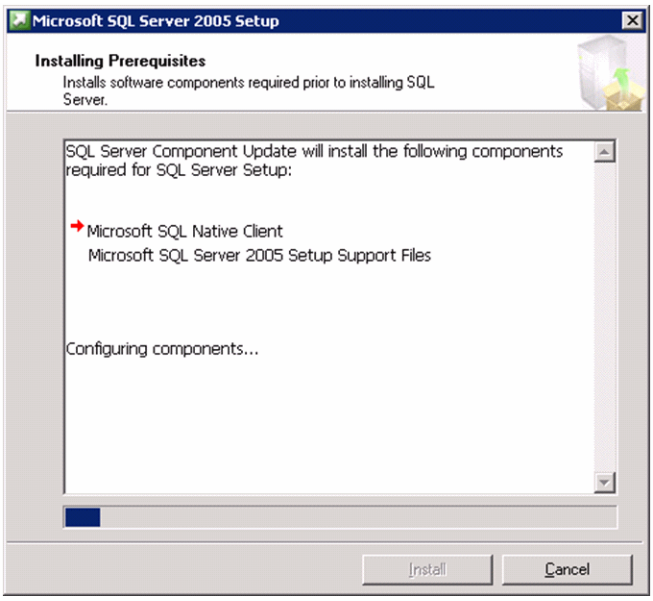

Microsoft SQL Server 2005 Setup
Installing Prerequisites
Installs software components required prior to installing SQL Server.
SQL Server Component Update will install the following components required for SQL Server Setup:
Microsoft SQL Native Client
Microsoft SQL Server 2005 Setup Support Files
Configuring components...
Install
Cancel

Figure 6-57 *Installing Prerequisites*

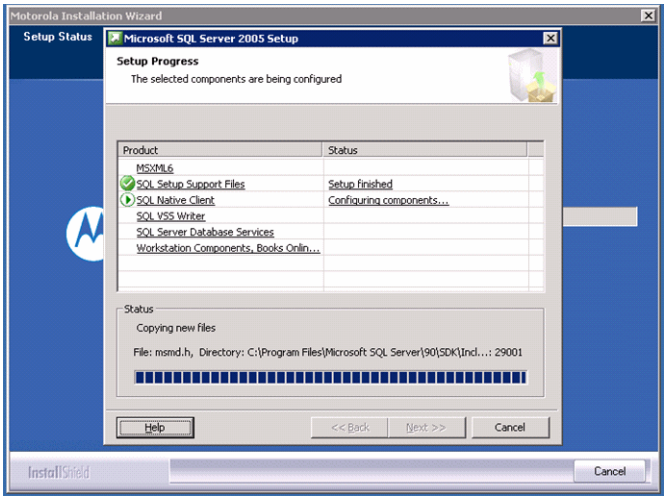


Figure 6-58 *SQL Server Setup*

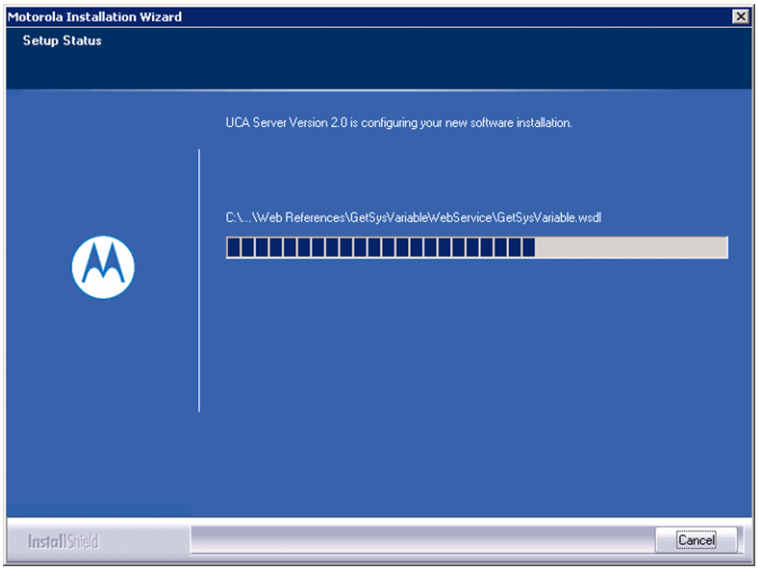


Figure 6-59 *SQL Server Setup Status*

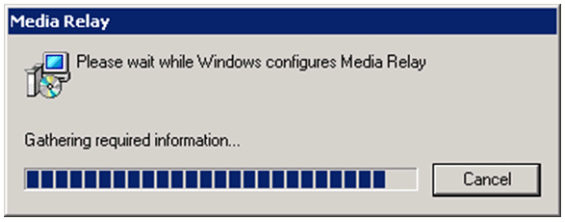


Figure 6-60 *Media Relay*

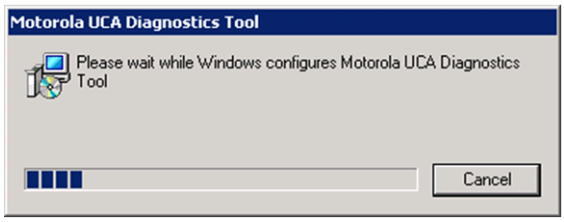


Figure 6-61 *UCA Diagnostics*

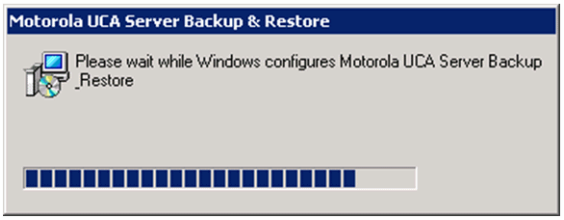


Figure 6-62 *UCA Server Backup & Restore*

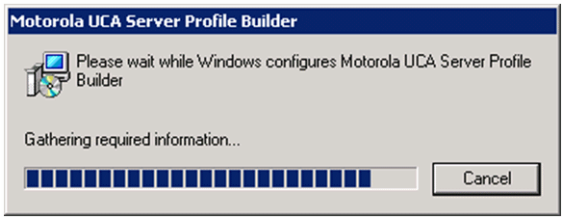

Motorola UCA Server Profile Builder
Please wait while Windows configures Motorola UCA Server Profile Builder
Gathering required information...
Cancel

Figure 6-63 *UCA Server Profile Builder*

21. Click **Yes** to start the console, or click **No**.

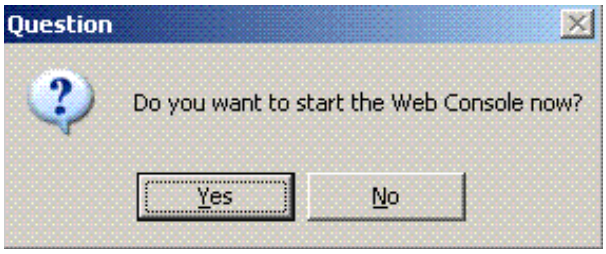


Figure 6-64 *Start Web Console*

22. If **Yes** is clicked to start the Web Console now the following Web Console window displays. See *Login to the UCA Server System on page 9-3* for login information.

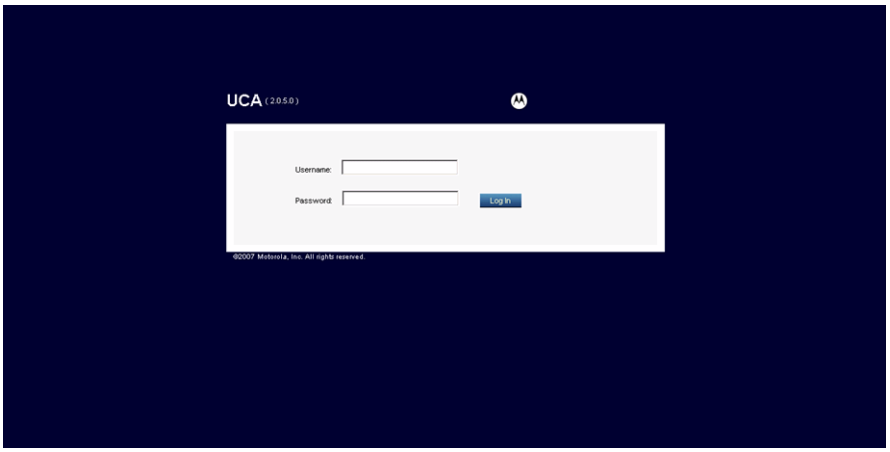


Figure 6-65 *WebConsole GUI Login*

IMPORTANT At the initial login to the Web Console (for details see *page 9-3*), it is required to use the default Web Console administrator *Username* and *Password*:

- *Factory Default Web Console Username*: *admin*
- *Factory Default Web Console Password*: *admin*

It is strongly suggested that the default administrator *Username* and *Password* be changed after the initial login. Admin Password can be changed on the UCA server console from System Management > Console Users > Console User Detail > Edit Console User Passwords.

Note that the *Username* and *Password* entered during the UCA server software installation process, (*step on page 6-13*) are for SQL database access and should not be mistaken with the default Web Console administrator Username and Password shown above.

23. The final dialog of the installation displays.

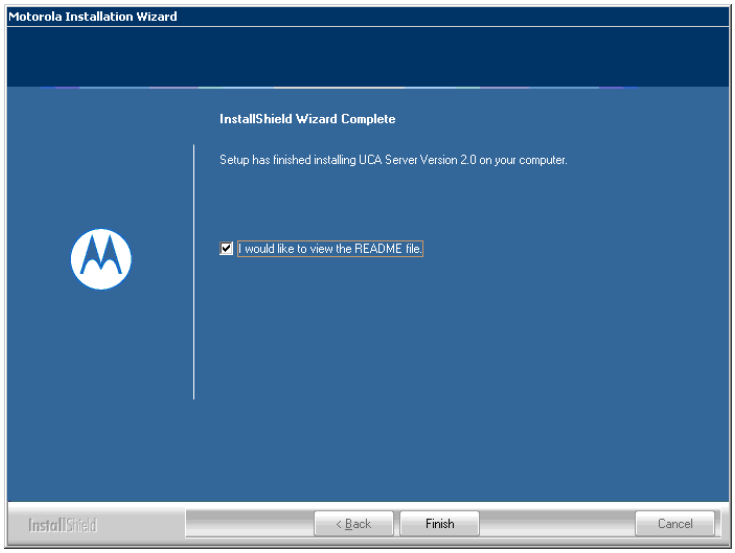


Figure 6-66 *InstallShield Wizard Complete*

24. To view the *Readme* document, leave the box checked and click **Finish**. To end the installation without viewing the *Readme* document, uncheck the box and click **Finish**.

25. After installing and launching the UCA server software, it is strongly recommended to validate that the server was successfully installed and is performing properly. For details see *Validate Successful Server Installation and Proper Performance on page 6-45*.

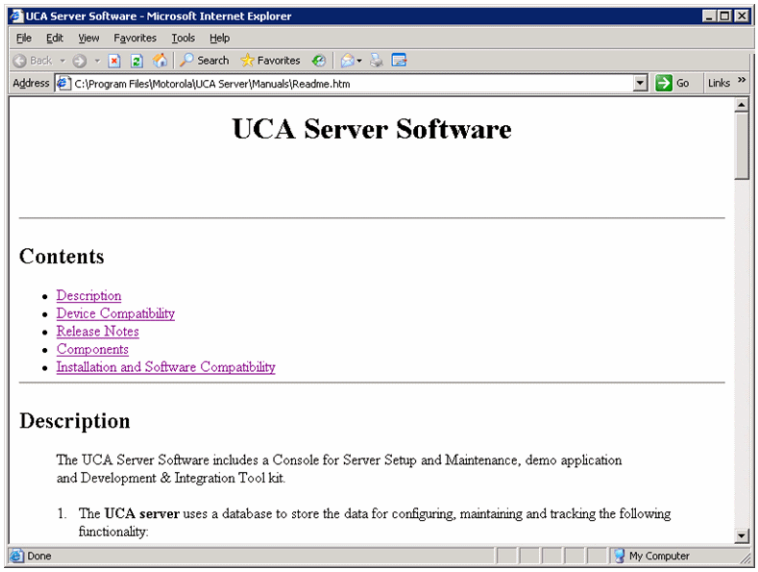

UCA Server Software - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Address C:\Program Files\Motorola\UCA Server\Manuals\Readme.htm
UCA Server Software
Contents
Description
Device Compatibility
Release Notes
Components
Installation and Software Compatibility
Description
The UCA Server Software includes a Console for Server Setup and Maintenance, demo application and Development & Integration Tool kit.
1. The UCA server uses a database to store the data for configuring, maintaining and tracking the following functionality:
Done
My Computer

Figure 6-67 *UCA Server Software Readme*

Uninstalling UCA Server Software

Two options exist for uninstalling the UCA server:

- Option A: Reload the same version of the UCA server software on a machine. This requires a complete uninstall. Actions 1, 2 and 3 (below) must be performed.
- Option B: Upgrade the UCA server software to a newer version. This requires only Action 1.

To uninstall the UCA server software, up to three actions may be required. Actions 1 and 2 include the removal (or modification) - from the *Add or Remove Programs* window - of the two programs listed below.

- Action 1: Remove the UCA server version x.x
- Action 2: Remove the UCA database component of SQL server 2005 (also known as the UCA Instance)**.**

  NOTE This action does not require the complete removal of Microsoft SQL Server 2005.

- Action 3: Remove the database files containing customer data. For additional details about this process, see *step* under *Action 2. Uninstalling the UCA Database Component (Instance) of the SQL Server 2005 on page 6-43*.

NOTE The UCA server must be removed prior to removing/modifying the SQL Server 2005 software.

Action 1. Uninstall the UCA Server

To uninstall the UCA server:

1. Select *Start* > *Settings* > *Control Panel* > *Add or Remove Programs*.
2. Select UCA Server Version x.x from the *Add or Remove Programs* list.
3. Click **Remove**.
4. A dialog displays for confirmation to uninstall the UCA server software.

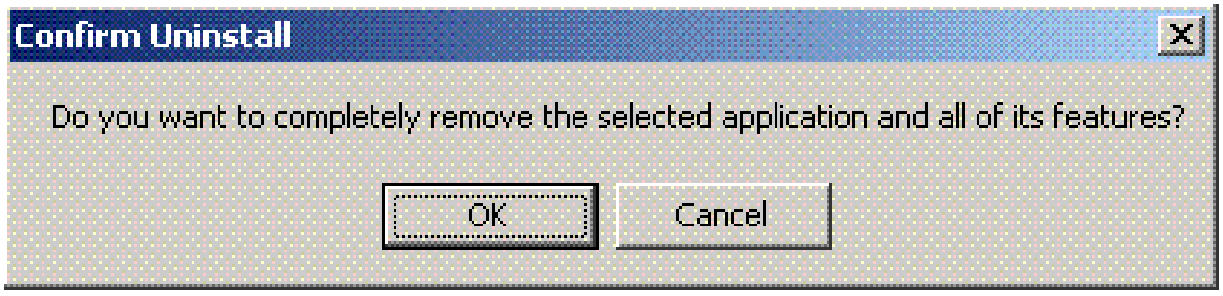


Figure 6-68 *Confirm Uninstall*

5. Click **OK** to proceed, or click **Cancel**.
6. Upon clicking **OK**, a dialog displays asking to keep or delete the UCA databases. Select the radio button corresponding to your choice, then click **Next**.

NOTE If any of the databases were populated with customer data, it is recommended that they remain on the PC for an upgrade.

7. The uninstall runs without further intervention.

8. Upon a successful uninstall, you are returned to the *Add or Remove Programs* screen.

IMPORTANT Option A: If you are reloading the same version of the UCA server, continue to Actions 2 and 3.

Option B: If you are upgrading the UCA server to a newer version STOP HERE! the uninstall process is complete.

Action 2. Uninstalling the UCA Database Component (Instance) of the SQL Server 2005

To uninstall only the UCA database engine instance without removing any other SQL instances:

1. Select *Start* > *Settings* > *Control Panel* > *Add or Remove Programs*.
2. On the *Add or Remove Programs* window, select Microsoft SQL Server 2005 and click **REMOVE**.
3. A dialog displays showing all installed SQL Server 2005 instances.

NOTE If this screen does not appear, cancel out of the un-install and repeat Step 2, but select **REMOVE**.

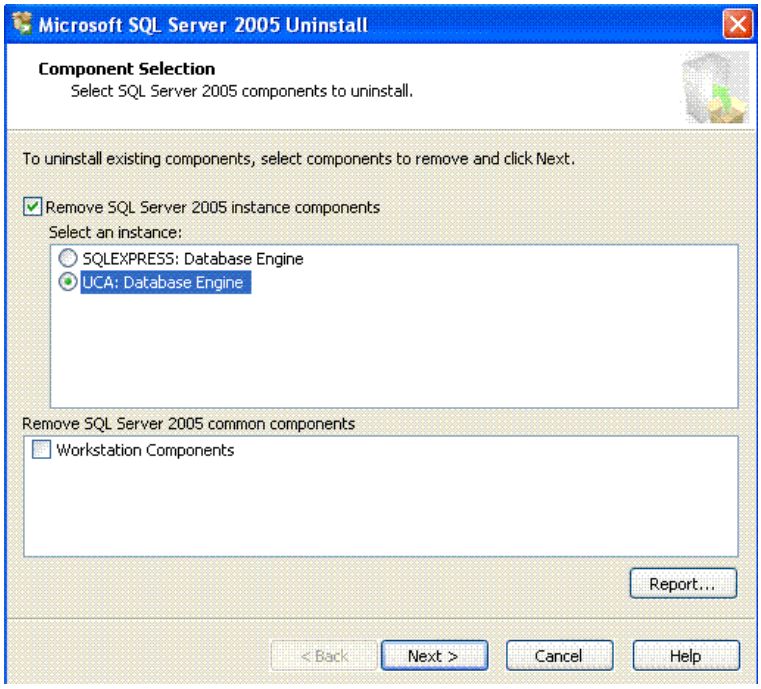


Figure 6-69 *UCA: Database Engine Uninstall*

Click the radio button to select the instance labeled *UCA: Database Engine.*

NOTE 1. Do not select/remove any other SQL Server 2005 instances that may be present. For example, in *Figure 6-69*, do not select/remove *SQLEXPRESS: Database Engine*.

2. Ensure the check box(es) in the section of the window labeled *Remove SQL Server 2005 common components* is not selected. For example, in *Figure 6-69*, *Workstation Components* must NOT be checked.

4. Click **Next** to proceed with the removal of the UCA database engine.
5. A window labeled *CONFIRMATION* and calling out the *DATABASE ENGINE: UCA* appears, confirming the uninstall of the UCA database engine.

6. Click **Finish**.
7. A Microsoft SQL Server 2005 dialog box displays as the instance of the UCA database engine uninstalls.
8. Upon its successful removal, the *Add or Remove Programs* window displays.

IMPORTANT Option A: If you are reloading the same version of the UCA server, continue to Action 3.

Option B: If you are upgrading the UCA server to a newer version STOP HERE! the uninstall process is complete.

Action 3. Validate the Removal of Database Files Containing Customer Data

If there were any errors during the UCA server portion of the uninstall process, manual cleanup may be required.

- Use Windows Explorer to navigate to C:\Program Files\Microsoft SQL Server\.
- There may be multiple MSSQL.x folders (MSSQL.1, MSSQL.2, etc.). To determine which folder represented the UCA instance, look in \MSSQL.n\MSSQL\data for UCA.mdf, UCA_log.ldf, SampleLookup.mdf and SampleLookup_log.ldf. These should be the only files in the folder. If so, delete them. If these database files are not found, the UCA uninstall successfully removed them.

Validate Successful Server Installation and Proper Performance

This section describes how to verify the installation and configuration of the UCA server.

Validate Proper Launch and Login to the UCA Server Web Console

To login to the UCA server (see *Login to the UCA Server System on page 9-3* for more information):

1. Launch the UCA server Web Console by double clicking the *Web Console for Server Setup and Maintenance* icon on the server desktop.
   Or
   Launch the IE browser and point the browser to one of the following URLs:
   - When logging in to the UCA server software locally, software is loaded on the computer being used and the following URL is applicable:
     http://localhost/Motorola/UCA/Client/DeviceManagement.asmx?op=DeviceRegistration
     Or
   - For remote login to the UCA server software, use the following URL and substitute the server IP address into the portion of the <*server IP address*> URL shown below. For example replace <*server IP address*> with *192.168.0.253* (use your relevant server IP address).
     http://<*server IP Address*>/Motorola/UCA/Client/DeviceManagement.asmx?op=DeviceRegistration
2. Login to the UCA server and Web Console as the administrator (use the short cut ';UCA Server Web Console' on the desktop).
   - Default Username: *admin*
   - Default Password: *admin*.

IMPORTANT At the initial login to the Web Console (for details see *page 9-3*), it is required to use the default Web Console administrator *Username* and *Password*:

- *Factory Default Web Console Username*: *admin*
- *Factory Default Web Console Password*: *admin*

It is strongly suggested that the default administrator *Username* and *Password* be changed after the initial login.

Note that the *Username* and *Password* entered during the UCA server software installation process, (*step on page 6-13*) are for SQL database access and should not be mistaken with the default Web Console administrator Username and Password shown above.

a. After a successful login, the System Status view of the UCA Web Console displays as shown in *Figure 9-2 on page 9-4*. This verifies that an administrator can login and view the status of the system.

IMPORTANT If the UCA Web Console does not display, the default Web page may be *stopped*. To *start* it do the following:

Go to *Start* > *Control Panel* > *Administrative Tools* > *Internet Information Services (IIS) Manager*. Click on the **+** next to the computer name under Internet Information Services. Expand the folder named Websites. If the default Web site is stopped, right click on it and select *Start*. Start the UCA Web Console (see *step 1* below).

Validate a CA50 Can Log In to the UCA Server

Simulate a CA50 login to the UCA server. Internet Explorer can be used to simulate a CA50 login, verifying that the system is set up properly for CA50 access.

1. Launch another instance of the IE browser and point the browser to the following **case-sensitive** URL:
   http://localhost/Motorola/UCA/Client/DeviceManagement.asmx?op=DeviceRegistration
   If the simulated CA50 login passes (the screen shown in *Figure 6-70* appears), an actual login using a CA50 should also be successful.
2. Execute the DeviceRegistration Web Service using the following values.

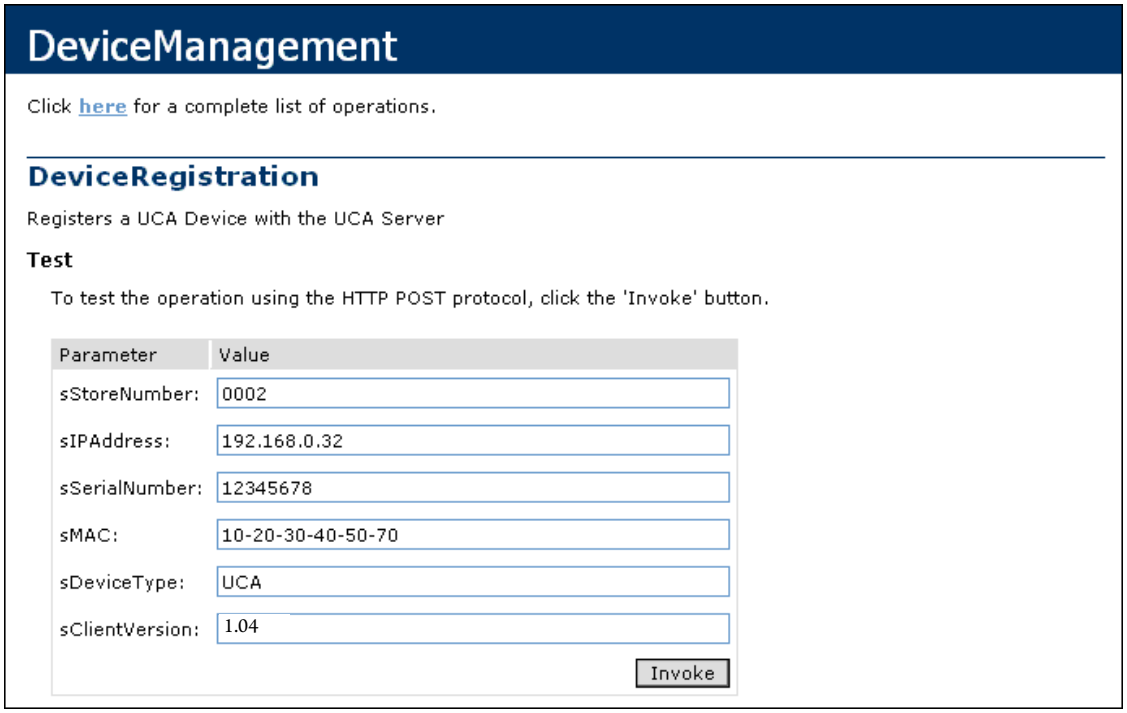


Figure 6-70 *Device Management - CA50 Login*

- sStoreNumber: *0002*
- sIPAddress: *192.168.0.32*
- sSerialNumber: 12345678
- sMAC: 10-20-30-40-50-70
- sDeviceType: UCA
- sClientVersion: 1.04

NOTE The value for *sClientVersion* should match the client level required by your version of the UCA server. This info can be found on the *System Status* view in the UCA Console. Look for the value assigned to *UCAClient Minimum Compatibility*.

3. Click **Invoke**.
4. A page with XML displays information that resembles, but is not identical to, the following:

```
<?xml version="1.0" encoding="utf-8" ?>
<string xmlns="http://Motorola.com/UCA/WS"><XMLURL><URL
URL="http://192.168.0.220/Motorola/UCA/Client/GetUser.aspx"/><DeviceConfig><cfg type="Time"
value="12/07/2007 13:09:04"/></DeviceConfig> </XMLURL></string>
```

5. Within the XML, search for a link that resembles **http://*Some_IP_Address*/Motorola/UCA/Client/GetUser.aspx**. This link is in bold type in this and the previous step to illustrate its location within the URL. This link is passed back to the CA50 during device registration to tell the device the URL used for login.
6. Launch another instance of the IE browser and copy and paste the link into the browser to display a page that says: *Scan barcode to log in*.
7. Enter a sample user ID that came with the installation sample data. For example, type *10007310* and press the **Enter** key. The browser should now display a blank page.
8. Right click on the blank page and choose *View Source*.
9. A big block of XML displays (approximately 7K of text data) the XML profile created by UCA server for user 10007310.
10. This information verifies a CA50 can log in to the UCA server.

Configuring Firewalls

The CA50 uses network communication to interact with the server and it is important to properly configure firewalls running on the server or on the network between the UCA server and the CA50. *Table 6-1* identifies what needs to be opened or allowed in pertinent firewalls.

Table 6-1 *Firewall Settings*

Process Name	Process Executable	Port / Direction	Direction
IIS	inetinfo.exe and/or w3wp.exe	80	Incoming
IP Blue MediaRelay	MediaRelay.exe	4747	Incoming

System Settings

Table 6-2 lists the system settings for CA50 Client/Server software. (See *User Profile Elements on page 7-4* for additional information about System Settings and their values.)

NOTE To enter these settings into the Web Console see *page 9-37*.

Table 6-2 *System Settings*

Setting	Description
Client.Backlight.Battery.Timeout	In minutes, how quickly the backlight times out while on battery
Client.Backlight.Level	Default backlight level on the device. Values are 0 - 4.
Client.Desktop.Link.DefaultScan	Default Desktop URL for bar code input. This setting allows the client to launch a customer business application with a bar code scanned while the user is at the client desktop. The sample data included references the UCASampleApp2. Note that this URL must be in the form of an http get request. There must be a "?" separating the Web page name from the parameters. The URL must also end with a parameter that doesn't have a value (i.e. "ProductBarcode="). The client software appends the scanned bar code data to the end of the URL if the last character of the URL is "=". Otherwise it does not include the bar code data and loads the page specified by the URL.
Client.NewPhoneRegRetryCount	Number of extensions to ask for when the first extension fails to register.
Client.PhoneRegRetryCount	Number of retries to register with the same extension.
Client.PhoneRegRetryInterval	Interval between successive retries.
Client.Phone.Ring	Device side path and filename for the default client ring tone for incoming phone calls.
Client.Volume.Receiver	Default receiver volume on the client. Values 0-6.
Client.Volume.Speaker	Default speaker volume on the client. Values 0-6.
Phone.Call.UnicastPacketRate	Unicast Call Packet Rate (0-40, in steps of 10).
Phone.Gateway.CallPark.ByGroup.Enable	Call Park by Group feature.
Phone.Gateway.CallPark.Enable	Call Park functionality is enabled on the PBX.
Phone.Gateway.FAC.CallPark	Feature Access Code for Call Park on the PBX.
Phone.Gateway.FAC.CallPick	Feature Access Code for Call Pick on the PBX.
Phone.Gateway.IPAddress	IP address for the phone gateway used for PBX communications.

[1] The Motorola UCA Maintenance Service must be restarted for changes to the System.Maintenance.DB.* and System.Maintenance.Session.* variables to take effect.

Table 6-2 *System Settings (Continued)*

Setting	Description
Phone.Gateway.Port	Port for the phone gateway used for PBX communications.
Phone.Gateway.Protocol.Type	The protocol and type of gateway with which the phone client attempts to communicate.
Phone.Gateway.SCCP.DeviceType	Device Type - set to 7960.
Phone.Peer2Peer.Enable	Enable direct CA50-to-CA50 IP based calls for CA50 devices. Device phone calls do not utilize a PBX if this is enabled.
Phone.Peer2Peer.Port	The port on which to make direct CA50-to-CA50 IP based calls.
Phonebook.Ranges.BaseRangeCount	The starting number of ranges to attempt to break a list into.
Phonebook.Ranges.MaxNonRangeSize	Maximum number of extensions to display as a single menu before breaking into ranges (multiple menus).
Phonebook.Ranges.MaxRangeSize	The maximum number of extensions to put in a range.
Phonebook.RecentCalls.Interval	Age in hours of past calls to consider as a recent call.
Phonebook.SelectableGroups.ShowAll	Determines whether or not the user associated with multiple groups appears in each department.
System.AllowDynamicWhenStaticFails	Determines whether or not to allow for a dynamic ext if the static fails to register.
System.DefaultLocation	Default external location ID to use on this server. In many cases this would be a store number.
System.Device.ActivityThreshold	The length of idle time (hours) before a device is considered in an unknown state.
System.Language.LanguageID	Language ID of the localized UCA Client (read only).
System.Logging.Level	Diagnostic trace logging. This setting controls how much detail is written to the trace log table (T_DebugTrace). All log entries of the specified level and greater are logged. Any log entries with lower priority levels are not logged. Possible Values: 0 - Debug 1 - Flow 2 - Status 3 - Warning 4 - Error Logging levels of 0 and 1 result in large amounts of logging data saved in the database. These levels are not recommended for normal system operation. They should be used with caution and only in cases where an issue was reported and is being monitored and investigated by Motorola.

[1] The Motorola UCA Maintenance Service must be restarted for changes to the System.Maintenance.DB.* and System.Maintenance.Session.* variables to take effect.

Table 6-2 *System Settings (Continued)*

Setting	Description
Phone.Gateway.Port	Port for the phone gateway used for PBX communications.
Phone.Gateway.Protocol.Type	The protocol and type of gateway with which the phone client attempts to communicate.
Phone.Gateway.SCCP.DeviceType	Device Type - set to 7960.
Phone.Peer2Peer.Enable	Enable direct CA50-to-CA50 IP based calls for CA50 devices. Device phone calls do not utilize a PBX if this is enabled.
Phone.Peer2Peer.Port	The port on which to make direct CA50-to-CA50 IP based calls.
Phonebook.Ranges.BaseRangeCount	The starting number of ranges to attempt to break a list into.
Phonebook.Ranges.MaxNonRangeSize	Maximum number of extensions to display as a single menu before breaking into ranges (multiple menus).
Phonebook.Ranges.MaxRangeSize	The maximum number of extensions to put in a range.
Phonebook.RecentCalls.Interval	Age in hours of past calls to consider as a recent call.
Phonebook.SelectableGroups.ShowAll	Determines whether or not the user associated with multiple groups appears in each department.
System.AllowDynamicWhenStaticFails	Determines whether or not to allow for a dynamic ext if the static fails to register.
System.DefaultLocation	Default external location ID to use on this server. In many cases this would be a store number.
System.Device.ActivityThreshold	The length of idle time (hours) before a device is considered in an unknown state.
System.Language.LanguageID	Language ID of the localized UCA Client (read only).
System.Logging.Level	Diagnostic trace logging. This setting controls how much detail is written to the trace log table (T_DebugTrace). All log entries of the specified level and greater are logged. Any log entries with lower priority levels are not logged. Possible Values: 0 - Debug 1 - Flow 2 - Status 3 - Warning 4 - Error Logging levels of 0 and 1 result in large amounts of logging data saved in the database. These levels are not recommended for normal system operation. They should be used with caution and only in cases where an issue was reported and is being monitored and investigated by Motorola.

[1] The Motorola UCA Maintenance Service must be restarted for changes to the System.Maintenance.DB.* and System.Maintenance.Session.* variables to take effect.

Table 6-2 *System Settings (Continued)*

Setting	Description
Phone.Gateway.Port	Port for the phone gateway used for PBX communications.
Phone.Gateway.Protocol.Type	The protocol and type of gateway with which the phone client attempts to communicate.
Phone.Gateway.SCCP.DeviceType	Device Type - set to 7960.
Phone.Peer2Peer.Enable	Enable direct CA50-to-CA50 IP based calls for CA50 devices. Device phone calls do not utilize a PBX if this is enabled.
Phone.Peer2Peer.Port	The port on which to make direct CA50-to-CA50 IP based calls.
Phonebook.Ranges.BaseRangeCount	The starting number of ranges to attempt to break a list into.
Phonebook.Ranges.MaxNonRangeSize	Maximum number of extensions to display as a single menu before breaking into ranges (multiple menus).
Phonebook.Ranges.MaxRangeSize	The maximum number of extensions to put in a range.
Phonebook.RecentCalls.Interval	Age in hours of past calls to consider as a recent call.
Phonebook.SelectableGroups.ShowAll	Determines whether or not the user associated with multiple groups appears in each department.
System.AllowDynamicWhenStaticFails	Determines whether or not to allow for a dynamic ext if the static fails to register.
System.DefaultLocation	Default external location ID to use on this server. In many cases this would be a store number.
System.Device.ActivityThreshold	The length of idle time (hours) before a device is considered in an unknown state.
System.Language.LanguageID	Language ID of the localized UCA Client (read only).
System.Logging.Level	Diagnostic trace logging. This setting controls how much detail is written to the trace log table (T_DebugTrace). All log entries of the specified level and greater are logged. Any log entries with lower priority levels are not logged. Possible Values: 0 - Debug 1 - Flow 2 - Status 3 - Warning 4 - Error Logging levels of 0 and 1 result in large amounts of logging data saved in the database. These levels are not recommended for normal system operation. They should be used with caution and only in cases where an issue was reported and is being monitored and investigated by Motorola.

[1] The Motorola UCA Maintenance Service must be restarted for changes to the System.Maintenance.DB.* and System.Maintenance.Session.* variables to take effect.

Table 6-2 *System Settings (Continued)*

Setting	Description
System.Login.MACAuthOnly	MAC Authorization is the only login option available. If enabled device MAC addresses are used to authenticate all devices and User ID's and PW are not used in the UCA Authentication Process.
System.Login.Token.GetPrimaryGroup	Login Page (URL) for device user to select primary group.
System.Login.Token.GetUserID	Login page (URL) for device user to enter User ID
System.Login.Token.GetUserPW	Login Page (URL) for device user to enter User PW/PIN.
System.Login.UserIDInputMethod	Device Side User ID Input Method - Scanner or Tumbler.
System.Maintenance.DB.CallParkExt.DaysBetween	Number of days between Call Park records cleaning.
System.Maintenance.DB.CallParkExt.FileAging	Number of exported metrics log files to leave on the disk.
System.Maintenance.DB.CallParkExt.RecordAging	Number of days to allow Call Park records to remain in the DB.
System.Maintenance.DB.DaysBetween [1]	Number of days between DB maintenance.
System.Maintenance.DB.DebugTrace.DaysBetween [1]	Number of days between Debug Trace table cleaning.
System.Maintenance.DB.DebugTrace.FileAging [1]	Number of exported debug trace log files to leave on UCA server hard disk. If this is set to zero, no debug trace logs are saved.
System.Maintenance.DB.DebugTrace.RecordAging [1]	Number of days to allow debug trace records to remain in the DB. This is the minimum number of days of debug trace data that is retained in the DB.
System.Maintenance.DB.Devices.DaysBetween [1]	Number of days between Device table cleaning.
System.Maintenance.DB.Devices.RecordAging [1]	Number of days to allow device records to remain in the DB.
System.Maintenance.DB.Export.Path [1]	Path to folder for DB maintenance exports. If the default path is changed to a different folder, the IIS_WPG user group must be given permissions to write files to it.
System.Maintenance.DB.Hour [1]	Hour of day to run DB maintenance. Values are 0 - 23. DB Maintenance can create potential server loading depending on the amount of data stored in the UCA database. DB Maintenance should be performed at off hours or when the system is least utilized.
System.Maintenance.DB.LocalUser.DaysBetween [1]	Number of days between cleaning local user records.
System.Maintenance.DB.LocalUser.RecordAging [1]	Number of days to allow local user records to remain in the DB.
System.Maintenance.DB.Metrics.DaysBetween [1]	Number of days between metrics records cleaning.
System.Maintenance.DB.Metrics.FileAging [1]	Number of exported metrics log files to leave on the disk.
System.Maintenance.DB.Metrics.RecordAging [1]	Number of days to allow metrics records to remain in the DB.

[1] The Motorola UCA Maintenance Service must be restarted for changes to the System.Maintenance.DB.* and System.Maintenance.Session.* variables to take effect.

Table 6-2 *System Settings (Continued)*

Setting	Description
System.Maintenance.Session.PollPeriod [1]	Time in minutes between checks for stale UCA user sessions.
System.Maintenance.Session.Timeout [1]	UCA session timeout in minutes.
System.Notif.CA50MediaRelay.Port	Command Port for the Media Relay Server.
System.Notif.MediaRelay.DefaultAudioEncoding	This setting may be used in future versions.
System.Notif.MediaRelay.IPAddress	IP address for the Media Relay Server.
System.Notif.MediaRelay.Port	Command Port for the Media Relay.
System.Notif.MCChannel.Address	Multicast address for System MC Notif Channel (Text Only).
System.Notif.MCChannel.Port	Multicast port for System MC Notif Channel (Text Only).
System.Profile.Path	Path to profile repository on server. This setting specifies a local path on the server's hard drive where the base XML profile files and profile schema are retrieved from by the UCA Authentication Process.
System.Profile.Schema.Path	Path to folder for XML schema to be used for profile validation.
System.Profile.Template.Path	Path to the folder that contains the user profile templates.
System.Profile.Workspace.Path	Path to the folder that contains profile builder wizard workspaces.
System.Resource.Path	The path for resource files (read only).
System.Tools.Stage	URL link to Motorola MSP Stage.
System.UCAServer.IPAddress	IP address of the UCA server.
System.Webservice.CallManager.URI	Base URI to Call Manager Web service (UCA phone book).
System.Webservice.CallPark.URI	Base URI to Call Park Web service.
WT.Base.MCAddress	Base multicast IP address used for walkie-talkie audio.
WT.Base.Port	Base port address used for walkie-talkie audio.
WT.ChannelIncrementMethod	Describes the method used for changing walkie-talkie channels. Values are: • Port • IP address

[1] The Motorola UCA Maintenance Service must be restarted for changes to the System.Maintenance.DB.* and System.Maintenance.Session.* variables to take effect.

UCA Server Security

The UCA server supports the following security related features:

- SSL Encryption
- Directory level password protection of sensitive Web services
- Encryption of server configuration files
- Encryption of passwords stored in the UCA database.

SSL (Secure Socket Layer)

Secure Sockets Layer is a protocol developed by Netscape for transmitting private documents via the Internet. SSL uses a cryptographic system that uses two keys to encrypt data: a public key known to everyone and a private, or secret key, known only to the recipient of the message. Both Netscape and Internet Explorer support SSL and many Web sites use the protocol to obtain confidential user information, such as credit card numbers. By convention, URLs that require an SSL connection start with https: instead of http:.

To enable SSL on the UCA server an SSL Certificate is required. The certificate can be purchase by trusted companies, such as VeriSign and others. For test purposes, Microsoft provides a utility called SelfSSL that can be used to generate SSL certificates for testing purposes only. The SelfSSL utility is part of the IIS Resource Kit available on the Microsoft Web site.

For the purpose of this discussion assume the use of SelfSSL to generate the certificate. Refer to the SelfSLL documentation for operational details.

Use SelffSSL to Generate the SSL Cert and Apply it to the Default Web site (UCA)

Generate a certificate with a common name of the IP address or domain name of the UCA server. The IP address or domain name is important and is used in later steps.

Make this Certificate Trusted by the Server

By default, this certificate is not trusted. To enable trust, install the certificate in the Trusted Root Certification Authorities Store.

Export the certificate to a file:

- In *IIS Manager*, right click on the default Web site.
- Select *Properties*.
- Select the *Directory Security* tab.
- In the *Secure Communications Section* choose *View Certificate*.
- The certificate you created applied with SelfSSL should show and have a red x through it indicating not trusted.
- Go to the *Details* tab and choose *copy to file*.
- Follow the wizard and choose *do not export private key, DER encoding and create file*.

Get the certificate into the Store:

- In Internet Explorer, choose *Tools* > *Internet Options*.
- Select the *Content* tab.
- Go to *Certificates*.
- In the *Certificates* dialog, go to the *Trusted Root Certification Authorities* tab.
- Click **Import**.
- Browse to the exported certificate created in the previous step.
- Select the exported certificate and import it into the store.

Give ASP.NET programs access to this certificate:

- Click on the Windows **Start** button.
- Select *Run...*.
- Type mmc (Microsoft Management Console).
- In mmc choose *File* > *Add/Remove Snap-in....*
- Click **Add**....
- Select the *Certificates* Snap-in.
- Click **Add**.
- Select *Computer account* and click **Next**.
- Select the *Local Computer* radio button and click **Finish**.
- Click **Close** on the underlying *Add Snap-in Dialog*
- Click **OK**.
- Expand the *Certificates* tree and select *Trusted Root Certification Authorities*.
- Right click on *Certificates - All Tasks* > *Import*.
- Use the wizard and import the previously exported certificate.

Modify the Web.configs and App.configs

Some of the UCA server config files contain Web references that must be modified to reflect the change when using SSL. If the web.config files are encrypted, they must be unencrypted before editing. See *Encryption/Decryption of the UCA Server Configuration Files on page 6-57* for details.

A Web reference in config files resembles the following:

```
<value>http://localhost/Motorola/UCA/Client/Authentication/Authentication.asmx</value>
```

Change Web references as follows:

- Change all references of http: to https:.
- Use the domain name or server IP address specified when creating the certificate in the Web reference links.

Modify config files as follows:

/Inetpub/wwwroot/Motorola/UCA/Client/web.config
/Inetpub/wwwroot/Motorola/UCA/UCAConsole/web.config
/Inetpub/wwwroot/Motorola/UCA/Samples/UCAMessaging/web.config
/Inetpub/wwwroot/Motorola/UCA/Samples/UCAPhoneExtensions/web.config
/Inetpub/wwwroot/Motorola/UCA/ServerManagement/web.config

/Program Files/Motorola/UCA Server/UCAMaintenanceService/UCAMaintWindowsService.exe.config

Modify System Settings

The following system settings must be modified to use https: rather than http:.

- Client.Desktop.Link.DefaultScan
- System.Login.Token.GetUserID
- System.Webservice.CallManager.URI
- System.Webservice.CallPark.URI

Modify User Profiles

All links specified in the user profiles must be modified to use *https:* rather than *http:*.

Enable SSL for the UCA Web Site

1. In *IIS Manager*, right click on the default Web site (UCA.
2. Select *Properties*.
3. Select the *Directory Security* tab.
4. In the *Secure Communications Control Section* choose *Edit*.
5. Select *Require Secure Channel*.
6. Select *Require 128-bit encryption*.
7. Click **OK**.

NOTE SSL must to be enabled on the UCA client application in order to communicate with the UCA server after enabling it on the server. See *Chapter 5, UCA Client Application on the CA50* for the UCA client application settings details.

Directory Level Password Protection for Sensitive Web Services

Some of the methods/actions available through the UCA Optional API Web Services are worthy of additional protection. This additional protection is provided by requiring a user name and password to execute the methods. The virtual directory that contains these Web Services is protected by Windows Basic authentication. In order to execute these methods from an external program, the user must supply a valid user name and password for the UCA machine. Because Windows Basic authentication sends user name and password to the server in the clear, it is recommended that SSL be enabled for the ServerManagement application, if it is to be used.

Encryption/Decryption of the UCA Server Configuration Files

To protect information found in the UCA server config files, sections within the files are encrypted during the install. This encryption uses the RSAProtectedConfigurationProvider. This provider uses the RSA public key encryption to encrypt and decrypt data. To ease in the decryption of the files for maintenance purposes, a few batch files are provided. They are found in C:\Program Files\Motorola\UCA Server\Utilities. Decrypt.bat and Encrypt.bat encrypts and decrypts the various config files associated with the UCA server. Decrypt.bat should not be deployed to a production environment for obvious reasons. In addition, keep in mind that because of the way the files are encrypted, they are tied to the machine on which they are encrypted. They cannot be unencrypted or used on another machine.

Encryption of Passwords Stored in the UCA Database

To protect passwords found in the UCA server database, they are encrypted before they are stored. The passwords are encrypted using Microsoft Cryptographic Service Provider.

CA50 Capabilities with an Offline UCA Server

If the UCA server is offline, a CA50 operates in the following ways.

- Voice (unicast) calls: Function without server support.[1]
- Walkie-talkie communication: Function without server support.
- Messaging and paging: Not supported until the server is online.
- Wireless bar code scanning: Not supported until the server is online.[2]

NOTE [1] For the CA50 to make voice calls while the server is down or offline, one phone book request from the CA50 to the UCA server had to be accessed from the server prior to the server going down. The CA50 reverts to the phone data provided by the last phone book request. When the phone book was previously viewed on the CA50, any portion of the phone book viewed on the CA50, and only this portion of the phone book, is cached on the CA50. If the UCA server is temporarily not accessible, the CA50 can access the cached phone book information.

[2] This assumes the business application resides on the UCA server. If it resides on a different addressable Web server in the enterprise, the business application functions properly.

Chapter 7 XML Profile for Configuration

Overview

This chapter defines what a profile is and what settings are included in the profile.

Valuable topics in this chapter include:

- XML Profile Overview (*page 7-2*)
- Delivery of the XML profile (*page 7-2*)
- XML profile layout (*page 7-3*)
- User profile section (*page 7-4*)
- Static menus section (*page 7-13*)
- Modifying phone book menu behavior (*page 7-17*)
- Descriptions of phone book Web service methods (*page 7-19*)
- Menu states (*page 7-22*)
- Key attribute definitions (*page 7-23*).

XML Profile Overview

The XML profile is a file used to configure the CA50. The XML profile can be modified by a developer to reflect a customer's desired CA50 device settings. Generally, the XML profile is loaded from the UCA server to the CA50. This loading occurs after the CA50 login into the UCA server.

NOTE 1. For a complete checklist outlining all the development and deployment activities required to implement a CA50 solution, see the *Development and Deployment Checklist on page 2-1*.

2. For a table listing the most commonly requested topics within this manual, see the Quick Startup Guide on the last page of the manual.

3. For XML samples, see *Base XML Profile Sample on page 7-25* and *Completed User XML Profile Sample on page 7-27*.

Delivery of the XML Profile to the CA50

There are two methods of delivery of the XML profile to the CA50 at login.

- Recommended method: The CA50 receives its XML profile from the UCA server during the login process.
- Alternate method: The XML profile resides on the CA50 in the form of an XML file. This method requires the use of a registry key for the client to find the location of the XML file on the CA50.

  The registry key is as follows:

  ```
  HKEY_LOCAL_MACHINE\SOFTWARE\Motorola\UCAClient\Params]
  "LocalProfileFile"="\application\ucaclient\bpasales.xml"
  ```

  When the XML file is found, the XML in the file provides the CA50 with enough information so that the device does not required communication with the UCA server.

XML Profile Layout

The XML profile contains a root element of <XMLProfile> which is comprised of two elements: <Profile> and <Menus>.

<Profile> Section of the XML Profile

The <Profile> portion of the XML profile contains several User Profile Elements that set the UCA client configuration settings.

The <Profile> portion of the XML profile is organized into the following categories/elements:

- User
- Audio
- Phone
- Walkie-Talkie
- DownloadFiles
- Desktop
- Device Config
- Alerts
- MetricsLog
- DebugLog
- Menu States.

<Menu> Section of the XML Profile

The <Menus> portion of the XML profile defines the static menu pages that the UCA client will display. These menus also provide the entry points to business applications and dynamic menu pages such the phone book.

The <Menus> portion of the XML profile is organized as a list of menus, it is not organized into categories like the <Profile> portion of the XML profile.

NOTE The structure of the static menu pages is identical to those used by the dynamic menu pages returned by the server as a result of Web service calls (i.e., the Call Manager Web service).

User Profile Section

NOTE See *page 7-25* and *page 7-27* for a sample Base XML Profile and Completed User XML Profile, respectively.

User Profile Elements

NOTE Several User Profile Elements values, such as *server.somecompany.com* and various values defined in this section come from the UCA Server System Settings database. See *Table 6-2 on page 6-48* for information about these entries.

Note that case matters when changing entries in the profile. *Error* does not equal *error* or *ERROR*.

There are eleven elements in the user profile section (<Profile>):

1. **User**
   The User element contains the basic user information: User ID, User Name, Store ID, Role ID, Role Name, Primary Group ID and Primary Group Name. These values are taken from the UCA Server System Settings database files.

2. **Audio**
   The Audio element specifies the default gain settings for the various audio paths. The audio paths that can be set are *Speaker* (used for walkie-talkie and notifications), *Receiver* (used for phone calls), *Mic* (microphone) and *HeadsetMic* (microphone in the headset). These settings can be saved as user preferences. The first time a user logs on, the default values are used. When a user logs off, the user's current settings are saved and are used the next time that user logs on. Valid entries are between 0 and 6.

3. **Phone**
   The Phone element specifies all settings necessary to configure the phone component. This includes the caller ID value that gets sent to the gateway and password that is used to register the assigned extension with the PBX. Currently, the only configurable value is Enable="Y" to allow phone usage, or Enable="N" to not allow phone usage.

   Additional parameters taken from the UCA console are:
   PhoneRegRetyCount,PhoneRegRetryInterval,NewPhoneRegRetryCount

   PBX - MAC, FACCALLPARK, FACCALLPICK, PACKETRATE

4. **Walkie-talkie**
   The walkie-talkie element specifies all necessary settings to set up the walkie-talkie component for default behavior with this profile. The first configurable value is Enable= "Y" to allow walkie-talkie usage, or Enable= "N" to not allow walkie-talkie usage. Multicast settings specify the multicast address to which the device should subscribe to receive walkie-talkie messages from other devices.
   There are three walkie-talkie channel settings:

- **Listen/Talk Channel**
  - Users can listen and talk on Listen/Talk channels.
  - A channel can be set to Listen/Talk by setting the *type* to '*T*'; in the user profile under the *WTT* > *Chlist* > *Ch* element.
  - The default channel is also Listen/Talk. A default channel is the channel used in normal mode and it is the channel that a user uses to talk. A channel can be set as a default channel by setting the *type* to '*Def*' in the *WTT* > *Chlist* > *Ch* element.

NOTE One Listen/Talk channel must be designated as the default channel for normal mode walkie-talkie communications.

- **Listen Only Channel**
  - Users can listen but cannot talk on Listen Only channels.
  - Users cannot manually select and enter single channel mode on Listen Only channels.
  - A channel can be set to listen only by setting the *type* to '*LO*' in the *WTT* > *Chlist* > *Ch* element.
- **Accessible (Non-listen) Channel**
  - By default, users cannot hear anything on this channel. The channel is used as a reserve channel for single channel mode.
  - A user must select this channel manually to access this channel.
  - Selecting this channel puts a user in *Single Channel* mode.
  - When selected in *Single Channel* mode, users can talk and listen only on this channel.
  - A channel can be set as *Accessible* (single channel only) by setting the *type* to '*SCO*' in the *WTT* > *Chlist* > *Ch* element.

There are four walkie-talkie communication modes. These different modes tell the device how to handle the channels to which it has access. The modes are:

- **Single Channel Mode**
  - Users can talk and listen only on this channel.
  - The only way to end *Single Channel* mode is to leave the channel manually.
- **Normal Mode**
  - Users listen on Listen/Talk channels and Listen Only channels.
  - Users talk on the default channel only.
  - *Normal* mode is exited when the user manually selects a channel other than the default talk channel, causing the device to enter *Single Channel* mode.
- **Scan Mode**
  - Users listen on Listen/Talk channels and Listen Only channels.
  - When a user presses **Talk**, the mode switches to *Single Channel* mode on the last Listen/Talk channel heard.
- **Do not disturb Mode**
  - Supports an extended period of time when a user wants/needs to be unreachable but is still able to use the business application. This may be required when a user takes a break or is given a task that does not require the communication feature.
  - In this mode the device does not receive any notifications or incoming walkie-talkie communication.

- **IP Address:** The multicast IP address to which the devices send/receive the WT messages. All the devices in the same subnet must subscribe to the same multicast IP address so they can listen and talk to other devices on the subnet.
- **Port:** The base port on which the device receives the text messages. All the other WT (voice message) channels listen on port numbers relative to this port.
- **Channel increment:** Defines how the WT channels are incremented relative to the base multicast address or base port. If the increment type is set to port, the device listens on the port number next to the base port number for Channel 1. For example: if the base port is 5000. If the channel increment type is set to port, the device listens on port 5001 for Channel 1, port 5003 for Channel 3, etc. Note that the device subscribes to only one Multicast IP address.

  The device sends the WT messages to the same multicast address (base multicast address), but listens to different channels on different ports.

  If the increment type is set to IP address, the device subscribes to multiple multicast addresses for the different channels. For example: if the base Multicast IP address is 239.192.2.2. and the channel increment type is set to IP address, the device subscribes to IP address 239.192.2.3 for Channel 1, 239.192.2.4 for Channel 2, etc.

  The device sends the WT messages to different multicast addresses for different channels, but listens to different channels on the same port (base port).

5. DownloadFiles

The UCAClient has the capability to download files from an http server to the CA50 device. This is useful when downloading custom .wav files to be used with notification or business applications.

Information about the files to download can be provided in the profile xml as follows:

```
<DownloadFiles DownloadAtStartUp="Y">
  <File HTTPURL="" PathOnDevice="" Overwrite="N"/>
  <File HTTPURL="" PathOnDevice="" Overwrite="N"/>
  <File HTTPURL="" PathOnDevice="" Overwrite="N" />
</DownloadFiles>
```

Attributes:

- DownloadAtStartUp="Y" - Select "Y" to download the files at user logon.
- HTTPURL - Source file location and name.
- PathOnDevice: - Destination (path on the CA50 device). The path on the CA50 device where the files are downloaded is "\Application\Download\".
- OverWrite="N" -Select "Y" to overwrite the existing file on the CA50 device, if applicable.

Example Usage:

```
<File HTTPURL="http://server.somecompany.com/Response.wav" PathOnDevice="" Overwrite="N" />
Source = "http://server.somecompany.com/Response.wav"
Destination = "\Application\Download"


<File HTTPURL="http://server.somecompany.com/Wav/Response.wav" PathOnDevice="\Wav\"
Overwrite="N" />
Source = "http://server.somecompany.com/Wav/Response.wav"
Destination = "\Application\Download\Wav\"
```

To download the configured files at a later time, after user logon:

Create a menu option to allow the user to download files upon user command.

Example:

```
<Menu ID="4" name="Applications" title="Applns">
  <entry ID="1" value="Download Files:S:L:DOWNLOADFILES"/>
  <entry ID="3" value="Price Check:S:S:http://server.somecompany.com/Sample.aspx"/>
</Menu>
```

The *Download files* option displays on the *Application* menu. When the option is selected, the files are downloaded onto the device.

6. **Desktop**
The Desktop element specifies actions that occur from the desktop while idle. These settings allow overrides of certain key actions. If a value is supplied in the URL attribute and the corresponding key is pressed, the browser loads that URL. The various supported types are:

- *DefaultScan* - occurs when a bar code is scanned from the desktop
- *UpHotLink* - occurs when **Up** is pressed from the desktop
- *DownHotLink* - occurs when **Down** is pressed from the desktop
- *FPTTHotLink* - occurs when the front **Walkie-Talkie** key is pressed from the desktop
- *SPTTHotLink* - occurs when the side **Walkie-Talkie** key is pressed from the desktop.
- *AutoLaunchHotLink* - occurs when the user logs into the CA50 device.

NOTE The user cannot use Walkie-Talkie if both FPTTHotLink and SPTTHotLink point to an application.

7. **DeviceConfig**
The DeviceConfig element is used to set the clock on the device and configure various other operational parameters such as timeouts. The *Time, BacklightLevel* and *BatteryBacklightTimeout* values are set from the server.

DeviceConfig values:

- CA50 Backlight - *ACBacklightTimeout* is the number of seconds the backlight stays on while the device is in the cradle.
  - Set *BatteryBacklightOnKeyPress* to 1 if the backlight should turn on when a key is pressed while on battery power; Set *BatteryBacklightOnKeyPress* to 0 if the backlight should not turn on when a key is pressed while on battery power.
  - Set *ACBacklightOnKeyPress* to 1 if the backlight should turn on when a key is pressed while on AC power; set *ACBacklightOnKeyPress* to 0 if the backlight should not turn on when a key is pressed while on AC power.
  - Set the *BacklightOnOff* to 1 to enable the backlight; set the *BacklightOnOff* to 0 to disable it.
  - Set the *BacklightEnableOnNotifMsg* to 1 to enable the backlight on receiving a notification message; set the *BacklightEnableOnNotifMsg* to 0 to disable it.
- *MenuTimeout* is the number of seconds that a menu is displayed without a key press.
- *WebServicePopupTime* is the number of seconds to wait on the response from the server before displaying a *Waiting for Server* message.
- *WebServiceResponseTimeout* is the amount of time a Web service call stays connected waiting for a response.

- *PopupErrorTimeout* is the amount of time in seconds an error message displays on the screen before it dismisses itself.
- *SilentModeTimeout* is the amount of time in seconds that the UCA client application mutes after **Mute** is pressed.
- *DNDModeTimeout* is the amount of time in seconds that the UCA client application stays in *Do not disturb* mode when that mode is entered.
- *PressNHoldTime* is ignored. It is currently not implemented.
- *NotifEnable* is set to "Y" to enable notifications; set *NotifEnable* to "N" to disable notifications.
- *NotifAllowOnCallActive* is set to "Y" to enable notifications while on a phone call; set *NotifAllowOnCallActive* to "N" to disable notifications while on a phone call.
- *CacheTimeToLive* is ignored. It is currently not implemented.
- Bar code scanning.
  - Set the *LogoutBarcode* value to allow the user to scan a bar code to logout. Set this value to "" to disable log off by bar code scanning. Note that, when enabled, scanning a bar code to log off works only when the user is logged in.
  - Set the *UCAExitBarcode* value to allow the user to scan a bar code to exit the UCA client application. Set this value to "" to disable exiting the UCA client application by bar code scanning. Note that, when enabled, scanning a bar code to exit the UCA client application works only when the user is logged in.

- Parameters that affect connectivity over an RF network.
  The XML profile parameters below establish the CA50's set points for quantifying an acceptable/unacceptable RF connection for communicating with the UCA server based on RF signal strength and recheck intervals. When these conditions are not met, the UCA client application un-registers the extension associated with making/receiving voice calls, disables the scanner, locks the keys and displays a *Network Connection Lost* message. Once these RF conditions are met again, the UCA client application re-establishes communication with the UCA server and re-enables all lost functionality.

  NOTE The Factory Default settings for these parameters should satisfy most implementation requirements. Modifying these parameters can negatively impact the performance of the system.

  - *BadRFSampleThreshhold* is the number of times the RF sample signal strength must be below the minimum signal strength before the *Network Connection Lost* message displays.
  - *GoodRFSampleThreshhold* is the number of times in a row the RF sample signal strength must be above the minimum signal strength before the bad sample count is reset.
  - *RFSampleTime* is the amount of time in seconds between sampling the RF signal strength.

    NOTE If *BadRFSampleThreshhold* is set to 3, *GoodRFSampleThreshhold* is set to 3 and *RFSampleTime* is set to 10 seconds, the earliest a *Network Connection Lost* message can display is after 3 bad samples in succession (3 * 10 seconds = 30 seconds). An addition time delay to display the message occurs if a few good samples are interspersed with bad samples.

  - *MinSignalStrength* is the minimum RF signal strength that is considered to be connected. Note that the numbers are negative and the larger the negative number, the weaker the signal. E.g., -70 is weaker than -60.
- *ACPowerCheckInterval* is the time in seconds between tests to see if the device is in a cradle.
- *NotifMsgTimeout* is the time in seconds for a notification message to display.
- Low battery indicator.

- *BatteryIconBeepEnable* is the parameter used to enable or disable the low battery indicator tone.
  Value '0' = disable
  Value '1' = enable.
- *DefaultLowBattWavFile* is used to indicate what sound (.wav file) to play when the battery goes below threshold level.
- *BatteryBeepInterval* sets the time interval in seconds between the beeps.
- PBX connectivity parameters.
- *PhoneRegistrationRetryCount* specifies the number of registration retries for the same number. The value can be zero or any positive number. When the value is zero there is no retry. When the value is a number greater than zero the number of retries equals the value of the number.

  NOTE This parameter does not apply to SCCP if the registration succeeds however the call manager does not return the number that was expected.

- *PhoneRegistrationRetryInterval* is the time in seconds between retries. The value must be greater than or equal to zero.
- *NewPhoneRegistrationRetryCount* specifies the number of times the client should request a new number when an attempt to register with a prior number fails (based on the *PhoneRegistrationRetryCount*).
- *NotifMsgVibrateCycleCount* specifies the number of times the device should vibrate on receipt of notification message.
- *NotifMsgVisibleCycleCount* specifies the number of times the device should show a visible indication on receipt of notification message.
- *AECForHeadset* enables echo cancellation for the headset mode, when selected. By default this option is disabled. This parameter is applied only to the headset mode and it is enabled for the handset at all times.

8. **Alerts**
   Use the alerts to configure device alerts for events. The indication types are defined as:

   - **Tone played**
   - **Visible indicator (LEDs lit in different colors)**
     - Blinking the LEDs
   - **Vibrate Indicator**

     For example **NotifCallActive** setting controls as what kind of alert is given to the user when a notification is sent to the user and the user is in an active call. It defines what kind of tone should be played, whether LED should blink or not. If so which LED should blink and how fast. Also controls whether the device should vibrate or not and if so how fast the device should vibrate.
   - **RcvCallNoCall**

     Defines the alert behavior when a phone call is received. The tone is picked up from the database; user cannot set the value for this attribute in the profile. Other attributes - Vibrate and Visible indicator can be set by the user in the profile.
   - **RcvWTTNoCall**

     Defines the alert behavior when receiving walkie-talkie audio data while not on an active call. User can set only the value for tone attribute in the profile.
   - **PTTNoCall**

     Defines the alert behavior when you press either **Walkie-Talkie** key when not on an active call. User can set only the value for tone attribute in the profile.

- **NotifNoCall**

  Defines the alert behavior when receiving a notification message while not on a phone call.

There are three alert resources:

1. Tone: A tone plays based on information from either the data within the notification message or from the profile data.
   - The device checks if the notification message has data to play the tone. If data is available, it plays the tone.
   - If the notification message has no data for tone, the device checks the user profile for this information. If the profile contains the tone information associated with NotifNoCall and the tone attribute is valid, that tone is played.
   - The tone selection information in the notification (message) takes precedence over the profile tone selection information.

   NOTE If the tone selection information in the message does not exist on the device, the profile tone is played.

2. LED: The LED flashes based on information from either the data within the notification message or from the user profile.
   - The device checks the notification message for the visible indicator data. If data is available, the LED flashes appropriately.
   - If the notification message has no visible indicator data, the device checks the user profile. If the profile contains the visible alert information associated with NotifNoCall, the corresponding LED flashes.
   - The visible indicator selection information takes precedence over the profile visible indicator selection information.
3. Vibrate: The device vibrates based on information from either the data within the notification message or from the user profile.
   - The device checks if the notification message has data regarding the vibrate indicator. If data is available, the device vibrates.
   - If the notification message has no data for vibrate indicator, the device checks the user profile. If the profile contains the vibrate alert information associated with NotifNoCall, the corresponding device vibrates.
   - The vibrate selection information takes precedence over the profile vibrate selection information.

- **NotifCallActive**

  This setting defines the alert behavior when receiving a notification message when on an active phone call.
  - The behavior of the alerts is the same as that of NotiNoCall, except that the settings in the profile take precedence over the message settings.

  NOTE The alerts for this event functions only if the NotifAllowOnCallActive (refer to device config - NotifAllowOnCallActive) is set to Yes.

- **AppAlertCallActive**

  When the device is running an application and an active call is in progress, this setting defines the *receiving a notification*, alert behavior. The allowed values for tone, visible and vibrate indicators are boolean values (whether or not the alerts are allowed in that state). If the tone, visible and/or vibrate indicator is allowed, the behavior of these alerts follows the same rules as NotifCallActive.
  - The device checks if any of the alerts' attributes (tone, visible, and vibrate) are enabled.
  - If the alert attributes are enabled, the attribute behavior applicable for NotifCallActive apply. The NotifCallActive attribute values appear in AppAlertCallActive (but are disabled).

- **AppAlertCallActive,** alert has three configurable options:
  - AllowTone
  - AllowVisibleIndicator
  - AllowVibrateIndicator
- All three attributes are Boolean. Permissible values are "Y" or "N".

> ***NOTE*** Attributes defined for NotifNoCall, NotifCallActive, and AppAlertCallActive apply only when the client setting Enable Notification is set to Yes. If this attribute is set to No, the notification is not sent to the device and none of the defined attributes will apply.

Visible Indicator attribute - possible values:

- RedSlowBlink - Red LED blinks once every second.
- RedMedBlink - Red LED blinks once every 1/2 second.
- RedFastBlink - Red LED blinks once every 1/10 second.
- RedOn - Red LED remains on.
- GreenSlowBlink - Green LED blinks once every second.
- GreenMedBlink - Green LED blinks once every 1/2 second.
- GreenFastBlink - Green LED blinks once every 1/10 second.
- GreenOn - Green LED remains on.
- AmberSlowBlink - Amber LED blinks once every second.
- AmberMedBlink - Amber LED blinks once every 1/2 second.
- AmberFastBlink - Amber LED blinks once every 1/10 second.
- AmberOn - Amber LED remains on.

Vibrate Indicator attribute - possible values:

- SlowVibrate - Device vibrates once every second.
- MedVibrate - Device vibrates once every 1/2 second.
- FastVibrate - Device vibrates once every 1/10 second.
- VibrateOn - Devices vibrates.

9. **MetricsLog**
   The MetricsLog element controls which possible metrics related activities are sent to the server for logging and to which server they are sent. Include the activities to be logged in the options list. Valid settings are "PTT" (Walkie- Talkie), "Device" (information specific to this device), "OutgoingCall" (information on calls to others) and "IncomingCall" (information on calls received). The amount of time between sends of this information can be modified by changing the CollectionInterval="60" statement. The value is in minutes.

   The CollectionIntervall is only for device and PTT metrics. The other metrics types are asynchronous and are sent up when either an outgoing or incoming phone call disconnects.

10. **DebugLog**
    The DebugLog element controls what types of log messages are recorded in the local <xx debug> log file (for more information, see *page 14-8*). Logging can be enabled for one or more categories and each can be set to a different level. The levels, in order of no messages to the most messages, are "Off" (no log messages), "Info" (device and server IP address), "Error" (information messages and errors), "Warn" (information, error and warning messages) and "Debug" (all messages). Note that there are log messages for start up logic even if all categories are set to “Off.” The UCAClient.reg file needs to be modified to turn off startup logging. The “LoggingEnabled” parameter under the [HKEY_LOCAL_MACHINE\SOFTWARE\Motorola\UCAClient\Params] key must be set to zero to completely disable logging.

11. **MenuStates**
    The MenuStates element controls what default soft key labels should be displayed and what their default actions should be for various states of the CA50. The values for the *State* attributes are discussed in *Static Menus Section on page 7-13* as they are very closely related to menu entries.

Static Menus Section

A sample static menu set is shown in the following example:

```
<Menus>
      <Menu ID="1" name="FrontMenu" title="Menu">
        <entry ID="1" value="Applications:M:L:Applications"/>
        <entry ID="2" value="WT Channels:M:L:FrontWTChannels" />
        <entry ID="3" value="PhoneBook:M:L:PhoneBook"/>
        <entry ID="4" value="Parked calls:M:L:CallPick"/>
        <entry ID="5" value="Key lock:S:L:KeyLock"/>
        <entry ID="6" value="Do Not Disturb:S:L:QuietMode:Y"/>
        <entry ID="7" value="Settings:M:L:Settings"/>
        <entry ID="8" value="Log Off:S:L:LogOffConfirm"/>
        <entry ID="9" value="Power Off:S:L:PowerOff"/>
        <entry ID="10" value="About:S:L:About"/>
      </Menu>
      <Menu ID="2" name="FrontAppMenu" title="Options">
        <entry ID="1" value="WT Channels:M:L:FrontWTChannels" />
        <entry ID="2" value="PhoneBook:M:L:PhoneBook"/>
        <entry ID="3" value="Parked calls:M:L:CallPick"/>
        <entry ID="4" value="Do Not Disturb:S:L:QuietMode:Y"/>
        <entry ID="5" value="Log Off:S:L:LogOffConfirm"/>
        <entry ID="6" value="Power Off:S:L:PowerOff"/>
      </Menu>
      <Menu ID="3" name="PhoneBook" title="Phn Bk">
        <entry ID="1"
value="Extensions:M:S:Range022.xml:http://server.somecompany.com/Motorola/UCA/Client/PhoneBook.asm
x/GetMenuExtRanges?sMenuType=Call&amp;sStoreNumber=%Location.ID%&amp;sListType=UCA&amp;sSortType=N
umber"/>
        <entry ID="2" value="Last
Names:M:S:Range024.xml:http://server.somecompany.com/Motorola/UCA/Client/PhoneBook.asmx/GetMenuExt
Ranges?sMenuType=Call&amp;sStoreNumber=%Location.ID%&amp;sListType=UCA&amp;sSortType=LastName"/>
        <entry ID="3"
value="Departments:M:S:Depts.xml:http://server.somecompany.com/Motorola/UCA/Client/phonebook.asmx/
GetMenuGroupList?sMenuType=Call&amp;sStoreNumber=%Location.ID%&amp;sListType=All&amp;sSortType=Nam
e"/>
        <entry ID="4" value="First
Names:M:S:Range023.xml:http://server.somecompany.com/Motorola/UCA/Client/PhoneBook.asmx/GetMenuExt
Ranges?sMenuType=Call&amp;sStoreNumber=%Location.ID%&amp;sListType=UCA&amp;sSortType=FirstName"/>
        <entry ID="5" value="Recent
Calls:M:S:Recent.xml:http://server.somecompany.com/Motorola/UCA/Client/PhoneBook.asmx/GetRecentCal
lList?sMenuType=Call&amp;sStoreNumber=%Location.ID%&amp;sListType=ALL&amp;sSortType=MostRecent"/>
      </Menu>
      <Menu ID="4" name="Applications" title="Applns">
        <entry ID="1" value="Price
Check:S:S:http://server.somecompany.com/Motorola/UCA/Samples/UCASampleApp/Sample.aspx"/>
        <entry ID="2"
value="PriceCheck2:S:S:http://server.somecompany.com/Motorola/UCA/Samples/UCASampleApp2/Default.as
px"/>
      </Menu>
      <Menu ID="5" name="FrontWTChannels" title="WT Channels">
        <entry ID="1" value="My WT Settings:S:L:SetChannel:5:Normal"/>
        <entry ID="2" value="Sales:S:L:SetChannel:3:Single"/>
      </Menu>
      <Menu ID="6" name="Settings" title="Settings">
        <entry ID="1" value="Backlight:M:L:Backlight"/>
        <entry ID="2" value="Backlight timer:M:L:BacklightTimer"/>
```

```
        <entry ID="3" value="Ring tone:M:L:RingTone"/>
        <entry ID="4" value="AEC for Headset:M:L:AECForHeadset"/>
      </Menu>
      <Menu ID="7" name="RingTone" title="Ring Tone">
        <entry ID="1" value="Basic:S:L:SetRing:\Application\Wav\ringtone basic.wav"/>
        <entry ID="2" value="Steeldrum:S:L:SetRing:\Application\Wav\Ringtone steeldrum.wav"/>
        <entry ID="3" value="Atmosphere:S:L:SetRing:\Application\Wav\Ringtone atmosphere.wav"/>
      </Menu>
      <Menu ID="8" name="DeskCallActive" title="Options">
        <entry ID="1" value="HangUp:S:L:PhoneHangUp"/>
        <entry ID="2" value="Transfer:M:L:TransferPhoneBook"/>
        <entry ID="3" value="Hold:S:L:PhoneHold"/>
        <entry ID="4" value="Park:M:L:CallPark"/>
        <entry ID="5" value="Applications:M:L:Applications"/>
        <entry ID="6" value="Settings:M:L:Settings"/>
      </Menu>
      <Menu ID="9" name="AppCallActive" title="Options">
        <entry ID="1" value="HangUp:S:L:PhoneHangUp"/>
        <entry ID="2" value="Transfer:M:L:TransferPhoneBook"/>
        <entry ID="3" value="Hold:S:L:PhoneHold"/>
        <entry ID="4" value="Park:M:L:CallPark"/>
        <entry ID="5" value="Settings:M:L:Settings"/>
    </Menu>
      <Menu ID="10" name="DeskQuiet" title="OnCall">
        <entry ID="1" value="Applications:M:L:Applications"/>
        <entry ID="2" value="Exit Quiet Mode:S:L:QuietMode:N"/>
      </Menu>
      <Menu ID="11" name="Backlight" title="Backlight">
        <entry ID="1" value="low:S:L:SetBkltLvl:0"/>
        <entry ID="2" value="medium-low:S:L:SetBkltLvl:1"/>
        <entry ID="3" value="medium:S:L:SetBkltLvl:2"/>
        <entry ID="4" value="medium-high:S:L:SetBkltLvl:3"/>
        <entry ID="5" value="high:S:L:SetBkltLvl:4"/>
      </Menu>
      <Menu ID="12" name="BacklightTimer" title="Bklgt Tmr">
        <entry ID="1" value="2 secs:S:L:SetBkltTimeout:2"/>
        <entry ID="2" value="5 secs:S:L:SetBkltTimeout:5"/>
        <entry ID="3" value="10 secs:S:L:SetBkltTimeout:10"/>
        <entry ID="4" value="20 secs:S:L:SetBkltTimeout:20"/>
      </Menu>
      <Menu ID="13" name="TransferPhoneBook" title="Phn Bk">
        <entry ID="1"
value="Extensions:M:S:Range022.xml:http://server.somecompany.com/Motorola/UCA/Client/PhoneBook.asm
x/GetMenuExtRanges?sMenuType=Transfer&amp;sStoreNumber=%Location.ID%&amp;sListType=UCA&amp;sSortTy
pe=Number"/>
        <entry ID="2" value="Last
Names:M:S:Range024.xml:http://server.somecompany.com/Motorola/UCA/Client/PhoneBook.asmx/GetMenuExt
Ranges?sMenuType=Transfer&amp;sStoreNumber=%Location.ID%&amp;sListType=UCA&amp;sSortType=LastName"
/>
        <entry ID="3"
value="Departments:M:S:Depts.xml:http://server.somecompany.com/Motorola/UCA/Client/phonebook.asmx/
GetMenuGroupList?sMenuType=Transfer&amp;sStoreNumber=%Location.ID%&amp;sListType=All&amp;sSortType
=Name"/>
        <entry ID="4" value="First
Names:M:S:Range023.xml:http://server.somecompany.com/Motorola/UCA/Client/PhoneBook.asmx/GetMenuExt
Ranges?sMenuType=Transfer&amp;sStoreNumber=%Location.ID%&amp;sListType=UCA&amp;sSortType=FirstName
"/>
      </Menu>
```

```
    <Menu ID="14" name="DeskCallOnHold" title="Options">
      <entry ID="1" value="HangUp:S:L:PhoneHangUp"/>
      <entry ID="2" value="Transfer:M:L:TransferPhoneBook"/>
      <entry ID="3" value="UnHold:S:L:PhoneResume"/>
      <entry ID="4" value="Applications:M:L:Applications"/>
    </Menu>
    <Menu ID="15" name="AppCallOnHold" title="Options">
      <entry ID="1" value="HangUp:S:L:PhoneHangUp"/>
      <entry ID="2" value="Transfer:M:L:TransferPhoneBook"/>
      <entry ID="3" value="UnHold:S:L:PhoneResume"/>
    </Menu>
      <Menu ID="16" name="AECForHeadset" title="AEC HS">
      <entry ID="1" value="Enable:S:L:SetAEC:1"/>
      <entry ID="2" value="Disable:S:L:SetAEC:0"/>
    </Menu>
</Menus>
```

Menu Elements

There can be any number of <Menu> elements contained in the <Menus> element. Each <Menu> can contain any number of <entry> elements. Each <Menu> element should be assigned sequential IDs. The <Menu> element name is the term by which the menu is referred to from another menu entry. The title is the display name in the title bar at the top of the CA50 display.

Menu entries should be listed in the order they are to display; their IDs should also be sequential. The value is interpreted as follows: 'Entry Text:Opt1:Opt2:data'

where:

'Entry Text' - is what gets displayed as the menu item to be selected

Opt1 - is M or S meaning (M)enu or (S)pecial option

Opt2 - is S or L meaning (S)erver or (L)ocal

data depends upon Opt1 and Opt2 as follows:

M:L:Menu Name - Specifies name of another locally stored static menu.

M:S:Menu.xml:URI - specifies name of dynamic menu and Web service URI to retrieve this dynamic menu which is cached locally.

S:L:Command[:options] - specifies an agent recognized action with possible options.

S:S:URL - URL to open with the UCA client application browser.

NOTE URIs or URLs with GET parameters need to use the "&" instead of just the "&" due to XML processing.

Changing the Ring Tone for the UCAClient Application

Step 1: Setting up the User Profile

The user profile contains a menu for ring tone options. To access ring tone options go to *Menu* > *Settings* > *Ring tone*. The portion of the *Ring tone* menu in the profile is listed as follows.

```
<Menus>
<Menu ID="1" name="FrontMenu" title="Menu">
.
.
<entry ID="6" value="Settings:M:L:Settings"/>
.
.
</Menu>
.
.
<Menu ID="7" name="Settings" title="Settings">
.
.
<entry ID="3" value="Ring tone:M:L:RingTone"/>
</Menu>
.
.
<Menu ID="7" name="RingTone" title="Ring Tone">
<entry ID="1" value="Basic:S:L:SetRing:\Application\Wav\ringtone basic.wav"/>
<entry ID="2" value="Steeldrum:S:L:SetRing:\Application\Wav\Ringtone steeldrum.wav"/>
<entry ID="3" value="Atmosphere:S:L:SetRing:\Application\Wav\Ringtone atmosphere.wav"/>
</Menu>
.
.
</Menus>
```

See *Menu Elements on page 7-15* for an explanation of value attributes in the *Ring tone* menu above.

In this example, the value attribute for the third entry in the *Ring tone* menu is "Atmosphere:S:L:SetRing:\Application\Wav\Ringtone atmosphere.wav". Therefore, "Atmosphere" displays as the third option in the *Ring tone* menu. If selected, a special local option is selected. The special local option is the command "SetRing" to "Application\Wav\Ring tone atmosphere.wav".

A customer could add more ring tones to this menu using the same format and providing the ring tone wave file is on the device.

Step 2: Changing the Ring tone on the CA50 via the UCAClient application

The ring tone selection can be seen via the *Menu* > *Settings* > *Ring tone* menu selection.

Once the atmosphere.wav file is selected as the ring tone, the user hears that sound when an incoming phone call occurs.

Modifying Phone Book Menu Behavior

The sample phone book menu as it appears in the *PhoneBook* menu (ID="3") and *TransferPhoneBook* menu (ID="13"), show the basic capabilities of the built-in phone book Web service. These phone book menus are for placing new calls and transferring existing calls, respectively. In general they probably parallel each other, but they can also be different to allow access to different extensions, whether making or transferring a call. These phone book menus can be easily modified by adding and/or removing menu entries.

There are three basic behavior tracks for the phone book:

- Group based (GetMenuGroupList & GetMenuExtListByGroup)
- Phone extension range based (GetMenuExtRanges & GetMenuExtListByRange)
- Recent call list (GetRecentCallList).

These tracks are all intended to make it easier to find a single desired extension out of a list of potentially hundreds of extensions. The group based track is a method of breaking up the list into a predefined hierarchy. The range based track is a method of dynamically breaking up the list into a hierarchy that changes as people log in and out. The recent call list is intended to show the extensions that are likely to be of most interest based upon call history.

Group Based Phone Book

When bringing up the phone book via the group based methods, the basic concept is organizing the phone book by the UCA server defined groups. Thus you start with a list of all groups with extensions associated to them. After a group is selected, all the extensions associated with that particular group will be displayed.

Phone Extension Range Based Phone Book

The basic concept of the range based phone book track is dividing a long list of extensions into dynamic ranges rather than groups. When a range is selected from the list, then all the extensions in that range are displayed. As an example, a store may have 100 extensions in use that range from 100 to 500 and are fairly evenly distributed. To break up this list of 100 extensions, you could divide this into 5 lists of 20 extensions that would be reasonably easy to navigate. So the ranges for this might look like the following: 100-180, 180-260, 260-340, 340-420, 420-500.

Phone Book Web Service

The behavior of each of the phone book entry points can be modified by changing the parameters included in the URL for each menu entry. The phone book Web service methods and their parameters are summarized in *Table 7-1*.

NOTE *%StoreLocation%* in the URL represents the store ID. This is replaced by the actual store ID (in System Settings) when the XML is sent to the device during login.

Table 7-1 *Web Service Methods Summary*

Web Service Method	sMenuType	sStoreNumber	sGroup	sListType	sSortType	sRangeStart	sRangeEnd
GetMenuGroupList	X	X		X	X		
GetMenuExtListByGroup	X	X	X	X	X		
GetMenuExtRanges	X	X		X	X		
GetMenuExtListByRange	X	X		X	X	X	X
GetRecentCallList	X	X		X	X		
GetUserListByGroup		X	X				

The phone book web service is accessible at this URL:

http://UCAInstallMachine/Motorola/UCA/Client/PhoneBook.asmx

Common Parameters

sMenuType - Specifies if this is a "call" or "transfer" menu.

Valid Values: call or transfer

sStoreNumber - Specifies the Store ID

sGroup - Group Token to filter by

sListType - This controls whether to list all extensions, only UCA extensions or only additional phone book entries (External).

Valid Values: All, UCA or External

NOTE Not all methods support all three options.

sSortType - This controls how the list is sorted and consequently how the entries appear.

Valid Values: Name, FirstName, LastName, Number, MostRecentl

NOTE - Not all methods support all options.
- The name options use the group name, the user's first/last/common name or the phone book description.
- The number options use the group number or the phone extension.

Descriptions of Phone Book Web Service Methods

GetMenuGroupList

This method returns a list of all groups that have phone book entries in them. The entries in the returned menu will retrieve a list of all the extensions in the selected group. All of the parameters are propagated to the entries. The URL used for the entries is derived from the system setting: System.Webservice.CallManager.URI.

Parameters

sListType

All supported (All, UCA or External)

sSortType

Name, FirstName, LastName or Number

NOTE The group list is sorted by either group number or group name based on the sort choice: one of the name sorts or the number sort. The sort type is passed on and applies to each group listing downstream.

Return Values

The method returns a valid XML menu structure. If there are no groups with extensions then it returns an empty menu that the client properly handles.

GetMenuExtListByGroup

This method returns a list of all extensions in the selected group. The entries in the returned menu will either place a new call or transfer an existing call.

Parameters

sListType

All supported (All, UCA or External)

sSortType

Name, FirstName, LastName or Number

Return Values

The method returns a valid XML menu structure. If there are no extensions in the selected group then it will return an empty menu that the client will properly handle.

GetMenuExtRanges

This method returns a list of extension ranges according to the parameters. The type of range is based on whether the sort order is by number or by a name. If the sort order is by number, then the ranges will be based on the actual phone extensions. If the sort order is by some name, then the ranges will be based on the first letter of the name portion specified.

Parameters

sListType

UCA

sSortType

Name, FirstName, LastName or Number

Return Values

The method returns a valid XML menu structure. If there are no extensions available according to the parameters then it will return an empty menu that the client will properly handle.

Remarks

- If the number of extensions to display is less than the specified threshold in the system setting Phonebook.Ranges.MaxNonRangeSize it is not broken into ranges. Instead, all extensions are returned.
- It does not make ranges that contain more extensions than the number specified in the system setting Phonebook.Ranges.MaxRangeSize.
- If there are enough extensions to break into ranges, it does not form fewer ranges than specified in the system setting Phonebook.Ranges.BaseRangeCount.

GetMenuExtListByRange

This method returns a list of all extensions in the selected range. The entries in the returned menu either place a new call or transfer an existing call.

Parameters

sListType

UCA

sSortType

Name, FirstName, LastName or Number

Return Values

The method returns a valid XML menu structure. If there are no extensions in the selected range then it returns an empty menu that the client properly handles.

GetRecentCallList

This method returns a list of the user's recent calls, both incoming and outgoing. The entries in the returned menu will either place a new call or transfer an existing call.

Parameters

sListType

All

sSortType

MostRecent

Return Values

The method returns a valid XML menu structure. If there are no extensions in the user's call history then it will return an empty menu that the client will properly handle.

Remarks

Since this is the user's call history; in order to use this phone book option, phone metrics must be enabled so that phone calls are reported to the UCA server to create the user's call history.

GetUserListByGroup

Use this to get the login IDs of the users who are currently logged into a particular group. The parameters required for this web service are:

Parameters:

a.sStoreNumber: Specifies the Store ID

sGroup: One of valid UCA group name.

Menu States

Defines the menus applicable for different menu states. For each state of the device (**AppCallActive,** indicates that the device is in a *call is active* and an *application is open in the browser* state) menu states define what is shown as the menu for the left soft key and the right soft key.

Following is a sample code snippet for the menu state *AppCAllActive*:

```
<MenuStates>
    <State ID="4" Name="AppCallActive" LSKey="Options:M:L:AppCallActive"
RSKey="End:S:L:PhoneHangUp" LPHKey="" RPHKey=""  FPTTKey="default"     SPTTKey="default" />
</MenuStates>
```

The format for representing a menu for LSKey and RSKey use is: [title to display]:M:L:[menu name]

In this example, the LSKey refers to the menu titled Options and the menu name is *AppCallActive*.

State = "AppCallActive"

Left Soft key (LSKey) displays options as a menu and the actual menu referenced is *AppCallActive***.**

Right Soft key (RSKey) displays end as a command and the actual command referenced is *PhoneHangUp.*

The name attribute identifies the client state as follows:

- DesktopNoCall - The user is at the desktop and not on a phone call.
- AppNoCall - The user is running a Web application and is not on a phone call.
- DesktopCallActive - The user is at the desktop and is on a phone call.
- AppCallActive - The user is running a Web application and is on a phone call.
- DesktopQuietMode- The user has entered quiet mode from the desktop.
- AppQuietMode - The user has entered quiet mode while running a Web application.
- DesktopCallOnHold-The user is on a call at the desktop and is on hold.
- AppCallOnHold - The user is on a call on hold while running a Web application.

Menus and actions can be assigned to keys for each of the client states. These menus are accessible to the user through the keys while the device is in that client state. Key and menu assignments must be thought through a bit. For example, for the "AppCallActive" client state is recommended to assign a menu to a key that presents a "Hang up" menu option.

As mentioned earlier, keys defined by the Web application take precedence over keys defined in the <MenuStates> area of the profile. The CA50 is a dual purpose device (communication and Web browser). The UCA Web developer must consider the <MenuStates> key assignments when designing a Web application for the CA50.

NOTE Examples of base profiles are found in the following directory in the default UCA server software installation: \Program Files\Motorola\UCA Server\Profiles.

Key Attribute Definitions

The key attribute definitions are listed below. See *Figure 3-1 on page 3-2* for key callouts.

- LSKey - Left soft key
- RSKey - Right soft key
- LPHKey - Left soft key push and hold (not implemented)
- RPHKey - Right soft key push and hold (not implemented)
- FPTTKey- Front **Walkie-Talkie** key
- SPTTKey- Side **Walkie-Talkie** key

Menu state attribute values that can be tied to keys are essentially menu entries. There are a few additional options that apply to specific attributes.

Options that apply to specific attributes are listed below:

- default - Perform default key action.

NOTE This is a valid option only for **Walkie-Talkie** keys.

- none or ' ' - No action when the key is activated.

NOTE This is a valid option for any key.

- Label: Opt1:Opt2:data - Label is shown as the soft key label on screen. When the soft key is pressed the specified action occurs.

NOTE This format is required for left and right soft keys.

- [Label]: Opt1:Opt2:data - The specified action occurs when the key is activated.

NOTE This format is used for push and hold (not implemented) and **Walkie-Talkie** keys where the label is optional since it is not displayed.

Setting up Walkie-Talkie Channels in the Base Profile

Setting up walkie-talkie channels involves creating or modifying two areas in the base profile: the WTT section and the menu section.

The WTT section below defines 3 channels: 1, 2 and 3. The values in this section are used to configure the VoIP subsystem with mulitcast address, port, allowed channels and types, etc.

```
<WTT Enable="Y" AllowScanMode="N">
    <Multicast IP="239.192.2.2" Port="5000" ChannelIncrement="Port" />
    <ChList>
      <Ch ID="1" type="LO" Name="Store Broadcast" SName="1" />
      <Ch ID="2" type="Def" Name="Sales" SName="1" />
      <Ch ID="3" type="SCO" Name="Front End" SName="1" />
    <ChList>
</WTT>
```

The menu excerpt section below create the UI displayed on the device that enables the user to switch between the channels defined above. The menu XML below creates a selectable menu entry labeled “WT Channels.” Then “WT channels” is selected from a menu, a sub-menu displays with two selectable items in the sub-menu: My WT Settings and Front End.

```
<Menu ID="5" name="SalesWTChannels" title="WT Channels">
    <entry ID="1" value="My WT Settings:S:L:SetChannel:2:Normal" />
    <entry ID="2" value="Front EndSettings:S:L:SetChannel:3:Single" />
</Menu>
```

When “Front End” is chosen from the menu, the CA50 client instructs the VoIP subsystem to se the WT channel to channel 3 (mote match to CH ID=”3” in the WTT section). When “My WT Settings” is chosen by the user, the CA50 client instructs the VoIP subsystem to set the WT channel to channel 2.

Base XML Profile Sample

A sample user profile portion is shown in the following example:

```
<Profile>
<User ID="from DB" Name="from DB" StoreID="from DB">
    <Role ID="0" Name="from DB"/>
    <Group ID="0" Name="from DB"/>
</User>
<Audio>
    <Volume type="Speaker" Value="0"/>
    <Volume type="Receiver" Value="0"/>
    <Volume type="Mic" Value="5"/>
    <Volume type="HeadsetMic" Value="5"/>
</Audio>
<Phone Enable="Y" Ext="from DB" CallerID="from DB" ExtPwd="0" MAC = "from DB">
    <Gateway IP="from DB" Port="from DB" Type="from DB" Protocol="from DB" DeviceType = "from DB"
    PacketRate ="from DB" FACCallPark="from DB" FACCallPick="from DB"/>
</Phone>
<WTT Enable="Y" AllowScanMode="N">
    <Multicast IP="239.192.2.2" Port="5000" ChannelIncrement="Port"/>
    <ChList>
      <Ch ID="1" type="LO" Name="Store Broadcast" SName="1"/>
      <Ch ID="3" type="SCO" Name="Sales" SName="3"/>
      <Ch ID="4" type="LO" Name="Quick Link" SName="4"/>
      <Ch ID="5" type="Def" Name="Front End" SName="5"/>
    </ChList>
</WTT>
<DownloadFiles DownloadAtStartUp="Y">
    <File HTTPURL="" PathOnDevice="" Overwrite="N"/>
    <File HTTPURL="" PathOnDevice="" Overwrite="N"/>
    <File HTTPURL="" PathOnDevice="" Overwrite="N" />
</DownloadFiles>
<Desktop>
    <Link type="DefaultScan" URL="from DB"/>
    <Link type="UpHotLink" URL=""/>
    <Link type="DownHotLink" URL=""/>
    <Link type="FPTTHotLink" URL=""/>
    <Link type="SPTTHotLink" URL=""/>
    <Link type="AutoLaunchHotLink" URL=""/>
</Desktop>
<DeviceConfig>
    <cfg type="Time" value="mm/dd/yyyy HH:MM:SS a/pm"/>
    <cfg type="BacklightLevel" value="from DB"/>
    <cfg type="BatteryBacklightTimeout" value="from DB"/>
    <cfg type="ACBacklightTimeout" value="60"/>
    <cfg type="BatteryBacklightOnKeyPress" value="1"/>
    <cfg type="ACBacklightOnKeyPress" value="1"/>
    <cfg type="BacklightOnOff" value="1"/>
    <cfg type="BacklightEnableOnNotifMsg" value="0"/>
    <cfg type="MenuTimeout" value="20"/>
    <cfg type="WebServicePopupTime" value="2"/>
    <cfg type="WebServiceResponseTimeout" value="20"/>
    <cfg type="PopupErrorTimeout" value="20"/>
    <cfg type="SilentModeTimeout" value="120"/>
    <cfg type="DNDModeTimeout" value="1800"/>
    <cfg type="PressNHoldTime" value="2"/>
    <cfg type="NotifEnable" value="Y"/>
    <cfg type="NotifAllowOnCallActive" value="Y"/>
    <cfg type="CacheTimeToLive" value="60"/>
```

```
    <cfg type="LogoutBarcode" value="UcaLogout"/>
    <cfg type="UcaExitBarcode" value="UcaClientExit"/>
    <cfg type="BadRFSampleThreshhold" value="3"/>
    <cfg type="GoodRFSampleThreshhold" value="3"/>
    <cfg type="RFSampleTime" value="10"/>
    <cfg type="MinSignalStrength" value="-77"/>
    <cfg type="ACPowerCheckInterval" value="5"/>
    <cfg type="NotifMsgTimeout" value="60"/>
    <cfg type="BatteryIconBeepEnable" value="1"/>
    <cfg type="DefaultLowBattWavFile" value="\\Application\\Wav\\AFFIRMATIVEFEEDBACK.WAV"/>
    <cfg type="BatteryBeepInterval" value="60"/>
    <cfg type="Language" value="from DB"/>
    <cfg type="PhoneRegRetryCount" value="from DB"/>
    <cfg type="PhoneRegRetryInterval" value="from DB"/>
    <cfg type="NewPhoneRegRetryCount" value="from DB"/>
    <cfg type="NotifMsgVibrateCycleCount" value="2" />
    <cfg type="NotifMsgVisibleCycleCount" value="2" />
    <cfg type="AECForHeadset" value="N"/>
</DeviceConfig>
<Alerts>
    <Alert Event="NotifNoCall" tone="" VisibleIndicator="" VibrateIndicator=""/>
    <Alert Event="NotifCallActive" tone="\Application\Wav\NewMsg.wav" volume="5"
    VisibleIndicator="" />
    <Alert Event="RcvCallNoCall" tone="from DB" VisibleIndicator="" VibrateIndicator="" />
    <Alert Event="RcvWTTNoCall" tone="\Application\Wav\wt.wav" />
    <Alert Event="PTTNoCall" tone="\Application\Wav\wt.wav" />
    <Alert Event="AppAlertCallActive" AllowTone="N" AllowVisibleIndicator="Y"
    AllowVibrateIndicator="Y"/>
    <Alert Event="AppAlertDND" AllowTone="N" AllowVisibleIndicator="Y" AllowVibrateIndicator="Y"/>
</Alerts>
<MetricsLog URI="http://server.somecompany.com/Motorola/UCA/Client/Metrics.asmx"
options="PTT,Device,OutgoingCall,IncomingCall" CollectionInterval="60"/>
<DebugLog>
    <Log category="WTT" level="Error"/>
    <Log category="Phone" level="Error"/>
    <Log category="Agent" level="Error"/>
    <Log category="Browser" level="Error"/>
    <Log category="Scanner" level="Error"/>
</DebugLog>
<MenuStates>
    <State ID="1" Name="DesktopNoCall" LSKey="Menu:M:L:FrontMenu" RSKey="Phn Bk:M:L:PhoneBook"
    LPHKey="" RPHKey="" FPTTKey="default" SPTTKey="default" />
    <State ID="2" Name="AppNoCall" LSKey="Options:M:L:FrontAppMenu" RSKey="Phn Bk:M:L:PhoneBook"
    LPHKey="" RPHKey="" FPTTKey="default" SPTTKey="default" />
    <State ID="3" Name="DesktopCallActive" LSKey="Options:M:L:DeskCallActive"
    RSKey="End:S:L:PhoneHangUp" LPHKey="" RPHKey="" FPTTKey="default" SPTTKey="default"/>
    <State ID="4" Name="AppCallActive" LSKey="Options:M:L:AppCallActive"
    RSKey="End:S:L:PhoneHangUp" LPHKey="" RPHKey="" FPTTKey="default" SPTTKey="default"/>
    <State ID="5" Name="DesktopQuietMode" LSKey="Exit DND:S:L:QuietMode:N" RSKey="" LPHKey=""
    RPHKey="" FPTTKey="default" SPTTKey="default"/>
    <State ID="6" Name="AppQuietMode" LSKey="Exit DND:S:L:QuietMode:N" RSKey="" LPHKey=""
    RPHKey="" FPTTKey="default" SPTTKey="default"/>
    <State ID="7" Name="DesktopCallOnHold" LSKey="Options:M:L:DeskCallOnHold"
    RSKey="End:S:L:PhoneHangUp" LPHKey="" RPHKey="" FPTTKey="default" SPTTKey="default"/>
    <State ID="8" Name="AppCallOnHold" LSKey="Options:M:L:AppCallOnHold"
    RSKey="End:S:L:PhoneHangUp" LPHKey="" RPHKey="" FPTTKey="default" SPTTKey="default"/>
</MenuStates>
</Profile>
```

Completed User XML Profile Sample

```
<?xml version="1.0" encoding="utf-8"?>
<XMLProfile>
    <Profile>
      <User ID="from DB" Name="from DB" StoreID="from DB">
        <Role ID="0" Name="from DB"/>
        <Group ID="0" Name="from DB"/>
      </User>
      <Audio>
        <Volume type="Speaker" Value="0"/>
        <Volume type="Receiver" Value="0"/>
        <Volume type="Mic" Value="5"/>
        <Volume type="HeadsetMic" Value="5"/>
      </Audio>
      <Phone Enable="Y" Ext="from DB" CallerID="from DB" ExtPwd="0" MAC = "from DB">
        <Gateway IP="from DB" Port="from DB" Type="from DB" Protocol="from DB" DeviceType = "from
DB" PacketRate ="from DB" FACCallPark="from DB" FACCallPick="from DB"/>
      </Phone>
      <WTT Enable="Y" AllowScanMode="N">
        <Multicast IP="239.192.2.2" Port="5000" ChannelIncrement="Port"/>
        <ChList>
          <Ch ID="1" type="LO" Name="Store Broadcast" SName="1"/>
          <Ch ID="3" type="SCO" Name="Sales" SName="3"/>
          <Ch ID="4" type="LO" Name="Quick Link" SName="4"/>
          <Ch ID="5" type="Def" Name="Front End" SName="5"/>
        </ChList>
      </WTT>
      <DownloadFiles DownloadAtStartUp="Y">
        <File HTTPURL="http://server.somecompany.com/Response.wav" PathOnDevice="" Overwrite="N" />
        <File HTTPURL="http://server.somecompany.com/Acknowledge.wav" PathOnDevice="" Overwrite="N"
        />
      </DownloadFiles>
      <Desktop>
        <Link type="DefaultScan" URL="from DB"/>
        <Link type="UpHotLink" URL=""/>
        <Link type="DownHotLink" URL=""/>
        <Link type="FPTTHotLink" URL=""/>
        <Link type="SPTTHotLink" URL=""/>
        <Link type="AutoLaunchHotLink" URL=""/>
      </Desktop>
      <DeviceConfig>
        <cfg type="Time" value="mm/dd/yyyy HH:MM:SS a/pm"/>
        <cfg type="BacklightLevel" value="from DB"/>
        <cfg type="BatteryBacklightTimeout" value="from DB"/>
        <cfg type="ACBacklightTimeout" value="60"/>
        <cfg type="BatteryBacklightOnKeyPress" value="1"/>
        <cfg type="ACBacklightOnKeyPress" value="1"/>
        <cfg type="BacklightOnOff" value="1"/>
        <cfg type="BacklightEnableOnNotifMsg" value="0"/>
        <cfg type="MenuTimeout" value="20"/>
        <cfg type="WebServicePopupTime" value="2"/>
        <cfg type="WebServiceResponseTimeout" value="20"/>
        <cfg type="PopupErrorTimeout" value="20"/>
        <cfg type="SilentModeTimeout" value="120"/>
        <cfg type="DNDModeTimeout" value="1800"/>
        <cfg type="PressNHoldTime" value="2"/>
        <cfg type="NotifEnable" value="Y"/>
```

```
        <cfg type="NotifAllowOnCallActive" value="Y"/>
        <cfg type="CacheTimeToLive" value="60"/>
        <cfg type="LogoutBarcode" value="UcaLogout"/>
        <cfg type="UcaExitBarcode" value="UcaClientExit"/>
        <cfg type="BadRFSampleThreshhold" value="3"/>
        <cfg type="GoodRFSampleThreshhold" value="3"/>
        <cfg type="RFSampleTime" value="10"/>
        <cfg type="MinSignalStrength" value="-77"/>
        <cfg type="ACPowerCheckInterval" value="5"/>
        <cfg type="NotifMsgTimeout" value="60"/>
        <cfg type="BatteryIconBeepEnable" value="1"/>
        <cfg type="DefaultLowBattWavFile" value="\\Application\\Wav\\AFFIRMATIVEFEEDBACK.WAV"/>
        <cfg type="BatteryBeepInterval" value="60"/>
        <cfg type="Language" value="from DB"/>
       <cfg type="PhoneRegRetryCount" value="from DB"/>
       <cfg type="PhoneRegRetryInterval" value="from DB"/>
       <cfg type="NewPhoneRegRetryCount" value="from DB"/>
      </DeviceConfig>
      <Alerts>
        <Alert Event="NotifNoCall" tone="" VisibleIndicator="" VibrateIndicator=""/>
        <Alert Event="NotifCallActive" tone="\Application\Wav\NewMsg.wav" volume="5"
VisibleIndicator="" />
        <Alert Event="RcvCallNoCall" tone="from DB" VisibleIndicator="" VibrateIndicator="" />
        <Alert Event="RcvWTTNoCall" tone="\Application\Wav\wt.wav" />
        <Alert Event="PTTNoCall" tone="\Application\Wav\wt.wav" />
        <Alert Event="AppAlertCallActive" AllowTone="N" AllowVisibleIndicator="Y"
AllowVibrateIndicator="Y"/>
        <Alert Event="AppAlertDND" AllowTone="N" AllowVisibleIndicator="Y"
AllowVibrateIndicator="Y"/>
      </Alerts>
      <MetricsLog URI="http://server.somecompany.com/Motorola/UCA/Client/Metrics.asmx"
options="PTT,Device,OutgoingCall,IncomingCall" CollectionInterval="60"/>
      <DebugLog>
        <Log category="WTT" level="Error"/>
        <Log category="Phone" level="Error"/>
        <Log category="Agent" level="Error"/>
        <Log category="Browser" level="Error"/>
        <Log category="Scanner" level="Error"/>
      </DebugLog>
      <MenuStates>
        <State ID="1" Name="DesktopNoCall" LSKey="Menu:M:L:FrontMenu" RSKey="Phn Bk:M:L:PhoneBook"
LPHKey="" RPHKey="" FPTTKey="default" SPTTKey="default" />
        <State ID="2" Name="AppNoCall" LSKey="Options:M:L:FrontAppMenu" RSKey="Phn Bk:M:L:PhoneBook"
LPHKey="" RPHKey="" FPTTKey="default" SPTTKey="default" />
        <State ID="3" Name="DesktopCallActive" LSKey="Options:M:L:DeskCallActive"
RSKey="End:S:L:PhoneHangUp" LPHKey="" RPHKey="" FPTTKey="default" SPTTKey="default"/>
        <State ID="4" Name="AppCallActive" LSKey="Options:M:L:AppCallActive"
RSKey="End:S:L:PhoneHangUp" LPHKey="" RPHKey="" FPTTKey="default" SPTTKey="default"/>
        <State ID="5" Name="DesktopQuietMode" LSKey="Exit DND:S:L:QuietMode:N" RSKey="" LPHKey=""
RPHKey="" FPTTKey="default" SPTTKey="default"/>
        <State ID="6" Name="AppQuietMode" LSKey="Exit DND:S:L:QuietMode:N" RSKey="" LPHKey=""
RPHKey="" FPTTKey="default" SPTTKey="default"/>
        <State ID="7" Name="DesktopCallOnHold" LSKey="Options:M:L:DeskCallOnHold"
RSKey="End:S:L:PhoneHangUp" LPHKey="" RPHKey="" FPTTKey="default" SPTTKey="default"/>
        <State ID="8" Name="AppCallOnHold" LSKey="Options:M:L:AppCallOnHold"
RSKey="End:S:L:PhoneHangUp" LPHKey="" RPHKey="" FPTTKey="default" SPTTKey="default"/>
      </MenuStates>
    </Profile>
    <Menus>
- <!--
```

Menu Entry values are formatted as follows:

'Entry Text:Opt1:Opt2:data'

where:

'Entry Text'- Displays as the menu item to be selected.

Opt1 - Is M or S for (M)enu or (S)pecial option.

Opt2 - Is S or L for (S)erver or (L)ocal.

data (with Opt1 and Opt2) is as follows:

M:L:Menu Name - Specifies the name of another locally stored menu.

M:S:Menu.xml:URI - Specifies the name of dynamic menu and Web service to retrieve the dynamic menu and is then cached locally under the label menu.xml.

S:L:Command[:options]-Specifies an agent recognized action with possible options.

S:S:URL - URL to send to browser. (This could also be handled as S:L:GoToURL:URL.)

```
-->
- <Menu ID="1" name="FrontMenu" title="Menu">
       <entry ID="1" value="Applications:M:L:Applications"/>
       <entry ID="2" value="WT Channels:M:L:FrontWTChannels" />
       <entry ID="3" value="PhoneBook:M:L:PhoneBook"/>
      <entry ID="4" value="Parked calls:M:L:CallPick"/>
      <entry ID="5" value="Key lock:S:L:KeyLock"/>
       <entry ID="6" value="Do Not Disturb:S:L:QuietMode:Y"/>
       <entry ID="7" value="Settings:M:L:Settings"/>
       <entry ID="8" value="Log Off:S:L:LogOffConfirm"/>
       <entry ID="9" value="Power Off:S:L:PowerOff"/>
       <entry ID="10" value="About:S:L:About"/>
     </Menu>
     <Menu ID="2" name="FrontAppMenu" title="Options">
       <entry ID="1" value="WT Channels:M:L:FrontWTChannels" />
       <entry ID="2" value="PhoneBook:M:L:PhoneBook"/>
      <entry ID="3" value="Parked calls:M:L:CallPick"/>
      <entry ID="4" value="Do Not Disturb:S:L:QuietMode:Y"/>
       <entry ID="5" value="Log Off:S:L:LogOffConfirm"/>
       <entry ID="6" value="Power Off:S:L:PowerOff"/>
     </Menu>
     <Menu ID="3" name="PhoneBook" title="Phn Bk">
       <entry ID="1"
value="Extensions:M:S:Range022.xml:http://server.somecompany.com/Motorola/UCA/Client/PhoneBook.asm
x/GetMenuExtRanges?sMenuType=Call&amp;sStoreNumber=%Location.ID%&amp;sListType=UCA&amp;sSortType=N
umber"/>
       <entry ID="2" value="Last
Names:M:S:Range024.xml:http://server.somecompany.com/Motorola/UCA/Client/PhoneBook.asmx/GetMenuExt
Ranges?sMenuType=Call&amp;sStoreNumber=%Location.ID%&amp;sListType=UCA&amp;sSortType=LastName"/>
       <entry ID="3"
value="Departments:M:S:Depts.xml:http://server.somecompany.com/Motorola/UCA/Client/phonebook.asmx/
GetMenuGroupList?sMenuType=Call&amp;sStoreNumber=%Location.ID%&amp;sListType=All&amp;sSortType=Nam
e"/>
       <entry ID="4" value="First
Names:M:S:Range023.xml:http://server.somecompany.com/Motorola/UCA/Client/PhoneBook.asmx/GetMenuExt
Ranges?sMenuType=Call&amp;sStoreNumber=%Location.ID%&amp;sListType=UCA&amp;sSortType=FirstName"/>
       <entry ID="5" value="Recent
Calls:M:S:Recent.xml:http://server.somecompany.com/Motorola/UCA/Client/PhoneBook.asmx/GetRecentCal
lList?sMenuType=Call&amp;sStoreNumber=%Location.ID%&amp;sListType=ALL&amp;sSortType=MostRecent"/>
```

```
      </Menu>
      <Menu ID="4" name="Applications" title="Applns">
        <entry ID="1" value="Price
Check:S:S:http://server.somecompany.com/Motorola/UCA/Samples/UCASampleApp/Sample.aspx"/>
        <entry ID="2"
value="PriceCheck2:S:S:http://server.somecompany.com/Motorola/UCA/Samples/UCASampleApp2/Default.as
px"/>
      </Menu>
      <Menu ID="5" name="FrontWTChannels" title="WT Channels">
        <entry ID="1" value="My WT Settings:S:L:SetChannel:5:Normal"/>
        <entry ID="2" value="Sales:S:L:SetChannel:3:Single"/>
      </Menu>
      <Menu ID="6" name="Settings" title="Settings">
        <entry ID="1" value="Backlight:M:L:Backlight"/>
        <entry ID="2" value="Backlight timer:M:L:BacklightTimer"/>
        <entry ID="3" value="Ring tone:M:L:RingTone"/>
        <entry ID="4" value="AEC for Headset:M:L:AECForHeadset"/>
      </Menu>
      <Menu ID="7" name="RingTone" title="Ring Tone">
        <entry ID="1" value="Basic:S:L:SetRing:\Application\Wav\ringtone basic.wav"/>
        <entry ID="2" value="Steeldrum:S:L:SetRing:\Application\Wav\Ringtone steeldrum.wav"/>
        <entry ID="3" value="Atmosphere:S:L:SetRing:\Application\Wav\Ringtone atmosphere.wav"/>
      </Menu>
      <Menu ID="8" name="DeskCallActive" title="Options">
        <entry ID="1" value="HangUp:S:L:PhoneHangUp"/>
        <entry ID="2" value="Transfer:M:L:TransferPhoneBook"/>
        <entry ID="3" value="Hold:S:L:PhoneHold"/>
        <entry ID="4" value="Park:M:L:CallPark"/>
        <entry ID="5" value="Applications:M:L:Applications"/>
        <entry ID="6" value="Settings:M:L:Settings"/>
      </Menu>
      <Menu ID="9" name="AppCallActive" title="Options">
        <entry ID="1" value="HangUp:S:L:PhoneHangUp"/>
        <entry ID="2" value="Transfer:M:L:TransferPhoneBook"/>
        <entry ID="3" value="Hold:S:L:PhoneHold"/>
        <entry ID="4" value="Park:M:L:CallPark"/>
        <entry ID="5" value="Settings:M:L:Settings"/>
    </Menu>
      <Menu ID="10" name="DeskQuiet" title="OnCall">
        <entry ID="1" value="Applications:M:L:Applications"/>
        <entry ID="2" value="Exit Quiet Mode:S:L:QuietMode:N"/>
      </Menu>
      <Menu ID="11" name="Backlight" title="Backlight">
        <entry ID="1" value="low:S:L:SetBkltLvl:0"/>
        <entry ID="2" value="medium-low:S:L:SetBkltLvl:1"/>
        <entry ID="3" value="medium:S:L:SetBkltLvl:2"/>
        <entry ID="4" value="medium-high:S:L:SetBkltLvl:3"/>
        <entry ID="5" value="high:S:L:SetBkltLvl:4"/>
      </Menu>
      <Menu ID="12" name="BacklightTimer" title="Bklgt Tmr">
        <entry ID="1" value="2 secs:S:L:SetBkltTimeout:2"/>
        <entry ID="2" value="5 secs:S:L:SetBkltTimeout:5"/>
        <entry ID="3" value="10 secs:S:L:SetBkltTimeout:10"/>
        <entry ID="4" value="20 secs:S:L:SetBkltTimeout:20"/>
      </Menu>
      <Menu ID="13" name="TransferPhoneBook" title="Phn Bk">
```

```
        <entry ID="1"
value="Extensions:M:S:Range022.xml:http://server.somecompany.com/Motorola/UCA/Client/PhoneBook.asm
x/GetMenuExtRanges?sMenuType=Transfer&amp;sStoreNumber=%Location.ID%&amp;sListType=UCA&amp;sSortTy
pe=Number"/>
        <entry ID="2" value="Last
Names:M:S:Range024.xml:http://server.somecompany.com/Motorola/UCA/Client/PhoneBook.asmx/GetMenuExt
Ranges?sMenuType=Transfer&amp;sStoreNumber=%Location.ID%&amp;sListType=UCA&amp;sSortType=LastName"
/>
        <entry ID="3"
value="Departments:M:S:Depts.xml:http://server.somecompany.com/Motorola/UCA/Client/phonebook.asmx/
GetMenuGroupList?sMenuType=Transfer&amp;sStoreNumber=%Location.ID%&amp;sListType=All&amp;sSortType
=Name"/>
        <entry ID="4" value="First
Names:M:S:Range023.xml:http://server.somecompany.com/Motorola/UCA/Client/PhoneBook.asmx/GetMenuExt
Ranges?sMenuType=Transfer&amp;sStoreNumber=%Location.ID%&amp;sListType=UCA&amp;sSortType=FirstName
"/>
      </Menu>
      <Menu ID="14" name="DeskCallOnHold" title="Options">
        <entry ID="1" value="HangUp:S:L:PhoneHangUp"/>
        <entry ID="2" value="Transfer:M:L:TransferPhoneBook"/>
        <entry ID="3" value="UnHold:S:L:PhoneResume"/>
        <entry ID="4" value="Applications:M:L:Applications"/>
      </Menu>
      <Menu ID="15" name="AppCallOnHold" title="Options">
        <entry ID="1" value="HangUp:S:L:PhoneHangUp"/>
        <entry ID="2" value="Transfer:M:L:TransferPhoneBook"/>
        <entry ID="3" value="UnHold:S:L:PhoneResume"/>
      </Menu>
      <Menu ID="16" name="AECForHeadset" title="AEC HS">
        <entry ID="1" value="Enable:S:L:SetAEC:1"/>
        <entry ID="2" value="Disable:S:L:SetAEC:0"/>
      </Menu>
    </Menus>
</XMLProfile>
```

(Page intentionally blank.)

Chapter 8 XML Profile Builder for Device Configuration

Overview

Profile Builder is a standalone (does not interface with the UCA database or any other UCA modules) Windows application installed with UCA software. This application provides an administrator user interface used to create new UCA XML profiles and to modify existing UCA XML profiles. This application is available as a separate installable, and can install on any Windows 2003 SE server or Windows XP machine.

NOTE *Profile Builder* does not interface with the UCA database or any other UCA modules.

Profile and Menu Contents

The UCA XML profile contains two sections: *Profile* and *Menu*.

The *Profile* sections contain the details for phone settings, Walkie-Talkie settings and settings related to the CA50 device and the UCA client. In addition, this section contains a *Menu States* section which defines the menu options in each operational state of the UCA client.

The *Menus* section contains the list of menu options that display on the CA50 device when the user logs in to the device.

Theory of Operation

Profile Builder allows a user to create and modify UCA XML profiles. This includes a *Menus* section of the profile constituting menus and menu states. The user can refer to the menus section during profile creation and profile modification.

Creation of UCA XML profiles using the *Profile Builder* application involves the following steps.

1. Create *Menus* section XML files constituting *Menu States* and *Menus* information.
2. Create a UCA XML profile by specifying other information and refer an existing *Menus* section XML file to the new profile.

UCA XML profiles in *Profile Builder* contain some parameters with values in the UCA database. The values are configured by the user using the UCA Server *WebConsole Graphical User Interface* and cannot be changed using *Profile Builder*. As a safeguard, *Profile Builder* protects these fields from modification and the field values are displayed in *Profile Builder* as *Auto* values.

The profiles created in *Profile Builder* are used by the UCA server. Upon user login to the CA50 device, the UCA server transfers the appropriate profile for the user to the CA50 device. Before transferring the profile to the device, the UCA server substitutes all *Auto* values in the profile with the database values.

XML Profile Layout

The *XMLProfile* has two components:

- *Profile*
- *Menus.*

XML Profile, Profile Section

The *Profile* portion contains *User Profile Elements* for UCA client configuration settings.

The *Profile* portion is organized into the following categories/elements:

- Phone Settings (*page 8-6*)
- Walkie-Talkie Settings (*page 8-7*)
- Desktop Settings (*page 8-11*)
- Volume Settings (*page 8-12*)
- Metrics & Logging (*page 8-13*)
- Device Settings (*page 8-14*)
- Client Settings (*page 8-15*)
- Alert Settings (*page 8-16*).

XML Profile, Menu Section

The *Menus* portion defines the static menu that the UCA client displays. These menus provide the entry points to business applications and dynamic menu pages such as the phone book.

The *Menus* portion is organized as a list of menus. Based on what is defined in the *Menu States*, different menus are assigned to left soft key and right soft key on the device for a particular state.

For example: In *DesktopNoCall* state the *SalesMenu* menu may be assigned for the left soft key and *PhoneBook* menu may be assigned for the right soft key.

NOTE The structure of the static menu pages is identical to the structure used by the dynamic menu pages and returned by the server after web service calls (i.e., the *Call Manager* web service).

XML Profile, Profile Section

Create New Profile

To create a new UCA XML profile, specify the new settings and save the profile. Specify the profile information, such as the profile name and the language:

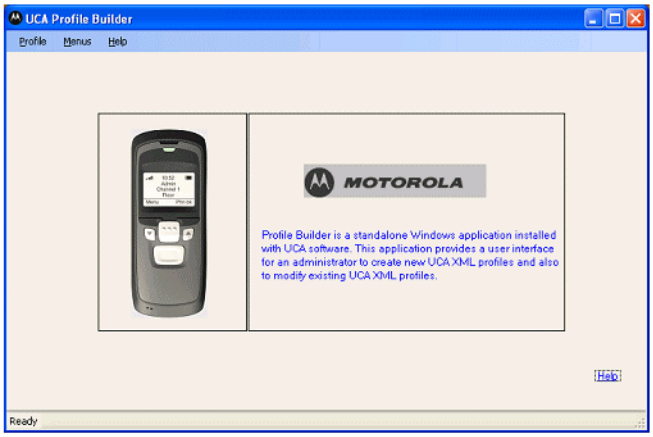

UCA Profile Builder
Profile
Menus
Help
MOTOROLA
Profile Builder is a standalone Windows application installed with UCA software. This application provides a user interface for an administrator to create new UCA XML profiles and also to modify existing UCA XML profiles.
Help
Ready

Figure 8-1 *Create New Profile*

1. Choose *Profile* > *New* to create a new profile. The *New Profile, Basic Details* page appears.

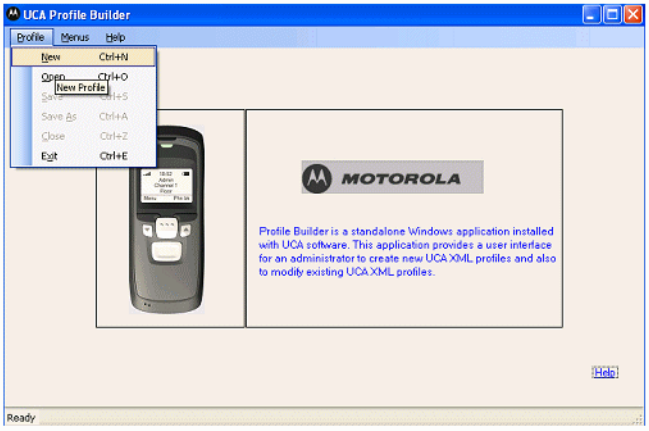


Figure 8-2 *New Profile, Basic Details*

2. Enter the profile name and select the language.

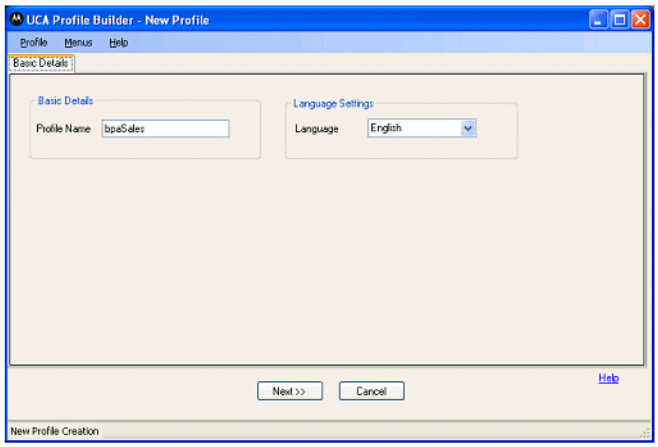


Figure 8-3 *Input Profile Basic Details*

NOTE English is the default language.The language list consists of English, French, Italian, Japanese, Korean, German and Spanish

3. Click **Next**. The *Menu Section* page appears. If creating a new profile, select the *Default Menu* or select *Choose Menu* to select from the existing profile (XML) files.

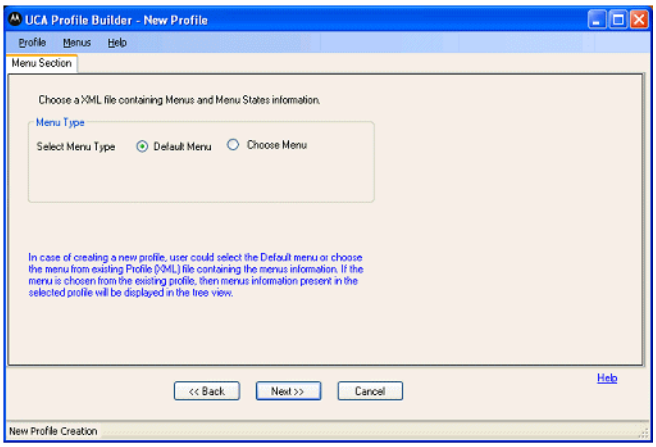

UCA Profile Builder - New Profile
Profile
Menus
Help
Menu Section
Choose a XML file containing Menus and Menu States information.
Menu Type
Select Menu Type
Default Menu
Choose Menu
In case of creating a new profile, user could select the Default menu or choose the menu from existing Profile (XML) file containing the menus information. If the menu is chosen from the existing profile, then menus information present in the selected profile will be displayed in the tree view.
<< Back
Next >>
Cancel
Help
New Profile Creation

Figure 8-4 *Menu Section Page*

4. The *Default Menu* option appears initially. To choose an existing menu, click the *Choose Menu* radio button. Click **Browse** to select an existing menu (an XML file containing menu information).

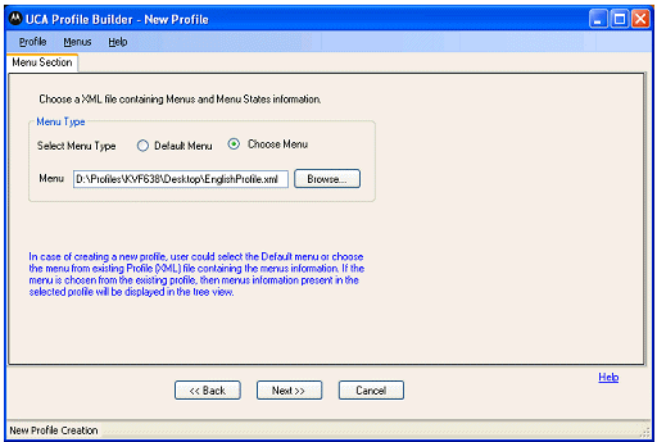

UCA Profile Builder - New Profile
Profile
Menus
Help
Menu Section
Choose a XML file containing Menus and Menu States information.
Menu Type
Select Menu Type
Default Menu
Choose Menu
Menu
D:\Profiles\KVF638\Desktop\EnglishProfile.xml
Browse...
In case of creating a new profile, user could select the Default menu or choose the menu from existing Profile (XML) file containing the menus information. If the menu is chosen from the existing profile, then menus information present in the selected profile will be displayed in the tree view.
<< Back
Next >>
Cancel
Help
New Profile Creation

Figure 8-5 *Configure Profiles*

Phone Settings

The Phone element specifies the settings necessary to configure the phone component. This includes the caller ID value (sent to the gateway) and the password used to register the assigned extension with the PBX.

Currently, the only configurable value is Enable="Y" to allow phone usage, or Enable="N" to not allow phone usage. The remaining parameters are disabled. They are populated from the database before the profile is sent from the UCA server to the device.

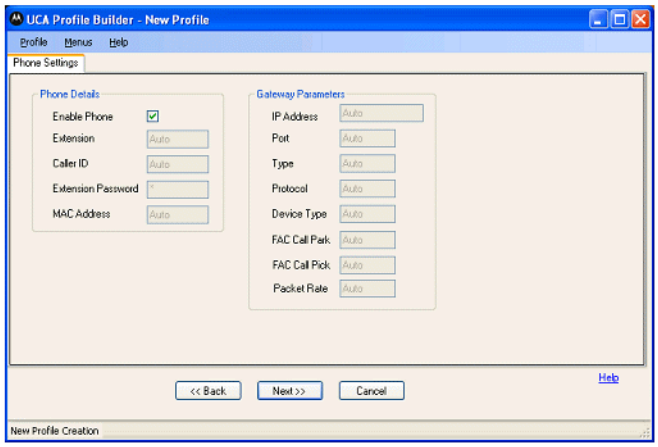


Figure 8-6 *Phone Settings*

Walkie-Talkie Settings

The Walkie-Talkie settings specifies the settings required to set up the default Walkie-Talkie communication.

To enable the Walkie-Talkie and *Scan* mode:

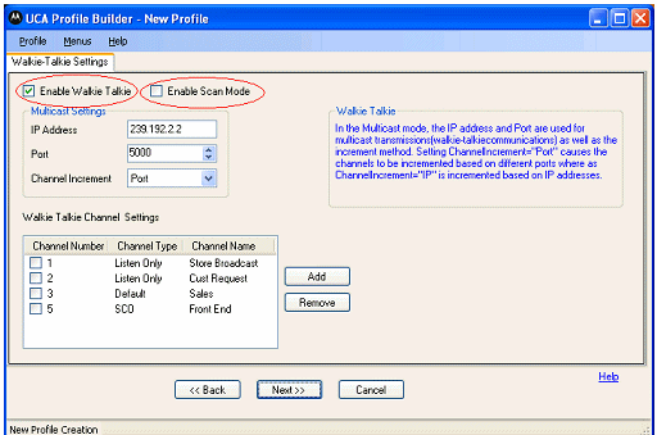


Figure 8-7 *Enable Walkie-Talkie*

1. Check the *Enable Walkie-Talkie* mode to enable the Walkie-Talkie communication. Un-check the *Enable Walkie-Talkie* mode to disable the Walkie-Talkie communication.
2. In *Scan* mode the device listens on *Listen/Talk* channels and *Listen Only* channels. Check the *Enable Scan* mode check box to have the device scan (listen) on the channels marked in the profile as *Listen/Talk* and *Listen Only*.

Multicast Settings

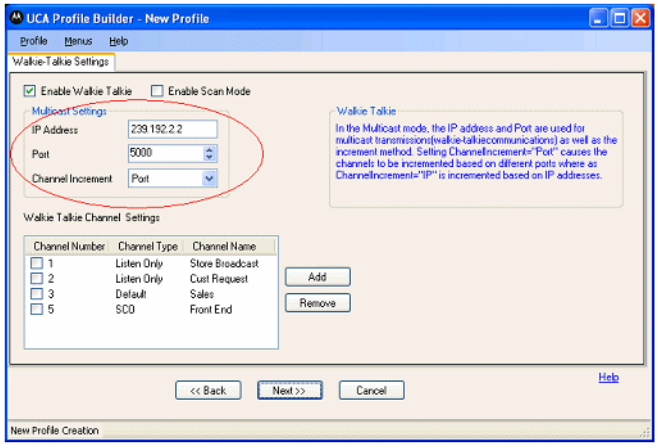


Figure 8-8 *Multicast Settings*

Multicast settings specify the multicast address which must be used to receive Walkie-Talkie (WT) messages.

Table 8-1 *Walkie-Talkie Multicast Settings*

Field	Description
IP Address	The multicast IP address is used to send/receive the WT messages to all the devices in the same subnet. All the devices (that need to listen and talk to other devices) must subscribe to the same multicast IP address. If the increment type is set to IP address, the device subscribes to multiple multicast addresses for the different channels. For example: See *Figure 8-8*, the base multicast IP address is 239.192.2.2. If the channel increment type is set to IP address, the device subscribes to IP address 239.192.2.3 for Channel 1, 239.192.2.4 for Channel 2, etc.
Port	The base port on which the device receives text messages. All the other WT (voice message) channels listen on port numbers relative to this port. If the increment type is set to Port, the device listens on the port number next to the base port number for Channel 1. For example: See *Figure 8-8*, the base port is 5000. If the channel increment type is set to port, the device listens on port 5001 for Channel 1, port 5003 for Channel 3, etc. The device subscribes to only one Multicast IP address. The device sends the WT messages to the base multicast address, but listens to different channels on different ports.
Channel increment	Defines how the WT channels are incremented relative to the base multicast address or base port.

Walkie-Talkie Channel Settings

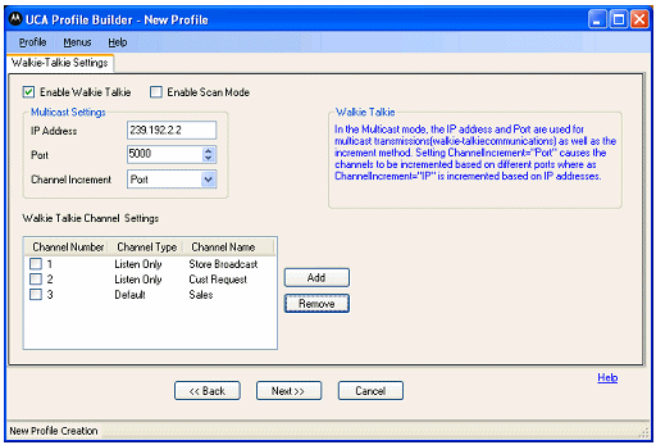


Figure 8-9 *Walkie-Talkie Channel Settings*

There are four types of Walkie-Talkie channels.

1. *Default*

   The channel assigned to the user on login, the user can talk and hear on this channel.

2. *Listen/Talk Channel*

   Users can listen and talk on *Listen/Talk* channels.

3. *Listen Only Channel*

   Users can listen to but cannot talk on *Listen Only* channels.

   Users cannot manually select and enter single channel mode on *Listen Only* channels.

4. *SCO (Single Channel Only)*

   By default, users cannot hear anything on this channel. This is a reserve channel for single channel mode.

   A user must select this channel manually to access it.

   Selecting this channel puts a user in *Single Channel* mode.

   When selected in *Single Channel* mode, users can talk and listen only on this channel.

NOTE One *Listen/Talk* channel must be designated as the default channel for normal mode Walkie-Talkie.
Steps in setting Walkie-Talkie values vary through the profile builder wizard.
Configure Walkie-Talkie channels and multicast settings.
There can be up to three Walkie-Talkie settings views.

Adding and Removing Channels

1. To add a new Walkie-Talkie channel, click **Add** to display the *Channel Details* screen.

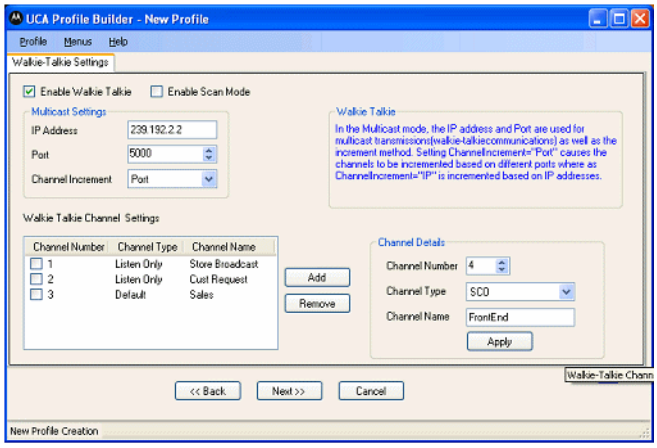


Figure 8-10 *Display Channel Details*

2. After adding the channel, the new channel appears in the Walkie-Talkie channel list, see the *New Channel* screen.

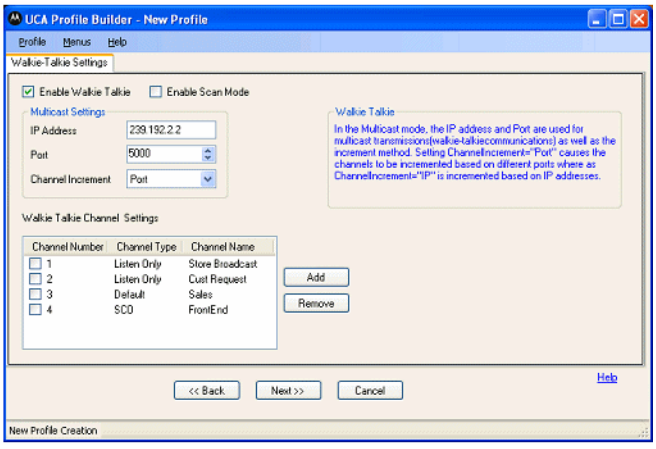


Figure 8-11 *New Channel*

3. Remove the channel by selecting the check box and clicking **Remove**.

Table 8-2 *Channel Settings*

Field	Description
Channel Number	Defines the channel number.
Channel Type	There are four available channel types. Only one channel can be selected as default.
Channel Name	*Channel Name* displays on the device menu. User can switch channels by selecting a channel name.

Download Files Settings

The Download File settings provides files to select from an HTTP server to download to the CA50 device. This feature is useful when downloading files that can be used with notification or business applications.

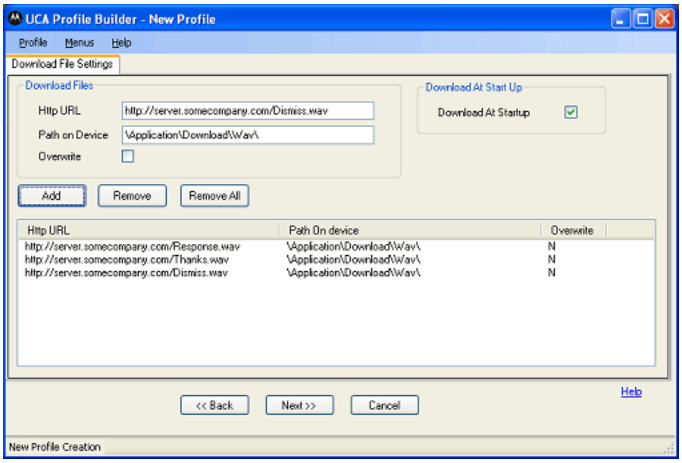


Figure 8-12 *Download Files Settings*

Table 8-3 *Download File Settings*

Field	Description
HTTP URL	Enter the source file path (location and name).
Path On Device	Enter the destination (path on the CA50 device). The path on the device is the location to which files are downloaded. The path on the device is “\Application\Download\”.
Overwrite	Check this check box to overwrite the existing file on the CA50 device, if it exists.
Download At StartUp	Check this check box to download the files soon after the user logs in.

Desktop Settings

When there are no active calls and there is no business application running on the device, the device state is *Desktop*. In this state actions occur from the desktop. However, overriding is allowed for some actions when the device is in *Desktop* mode. If a value is supplied in the URL attribute and the corresponding key is pressed (in *Desktop* mode) the browser loads that URL.

Set URL values for different keys (key functions). When on the desktop, pressing any of these keys invokes the corresponding URL. The URLs can be used to point to business applications.

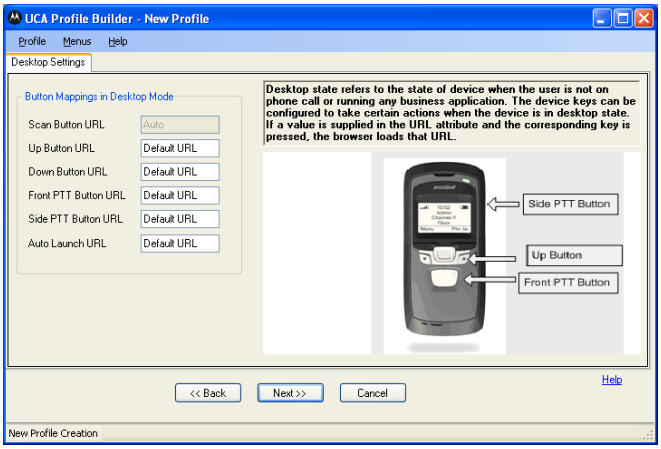

UCA Profile Builder - New Profile
Profile
Menus
Help
Desktop Settings
Button Mappings in Desktop Mode
Scan Button URL
Auto
Up Button URL
Default URL
Down Button URL
Default URL
Front PTT Button URL
Default URL
Side PTT Button URL
Default URL
Auto Launch URL
Default URL
Desktop state refers to the state of device when the user is not on phone call or running any business application. The device keys can be configured to take certain actions when the device is in desktop state. If a value is supplied in the URL attribute and the corresponding key is pressed, the browser loads that URL.
Side PTT Button
Up Button
Front PTT Button
Help
<< Back
Next >>
Cancel
New Profile Creation

Figure 8-13 *Desktop Settings*

Table 8-4 *Desktop Settings*

Field	Description
Scan Button URL	Business Application URL to launch when a bar code is scanned from the desktop.
Up Button URL	Business Application URL to launch when the **Up** arrow key is pressed from the desktop.
Down Button URL	Business Application URL to launch when the **Down** arrow key is pressed from the desktop.
Front PTT Button URL	Business Application URL to launch when the front **Walkie-Talkie** (FPTT) key is pressed from the desktop.
Side PTT Button URL	Business Application URL to launch when the side **Walkie-Talkie** (SPTT) key is pressed from the desktop.
Auto Launch URL	Business Application URL to launch as soon as the user logs into the UCAclient (without any user action).

Volume Settings

The volume settings define the default values for the audio paths (headset, microphone, speaker and receiver). The default values are used for the first time login, however once the settings are changed, the new values are saved and are used with the next time log-on.

Use the *Volume Settings* screen to set the volume level for audio output controls. Allowed values are between 0 and 6.

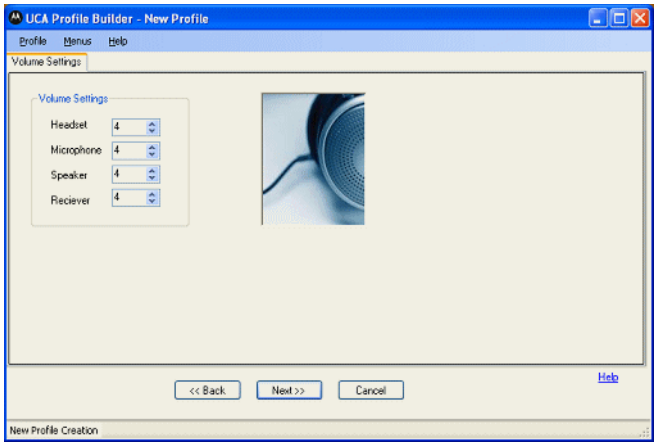


Figure 8-14 *Volume Settings*

Metrics & Logging

Metrics

The metrics section specifies the required device metrics, what is sent to the server and defines the interval.

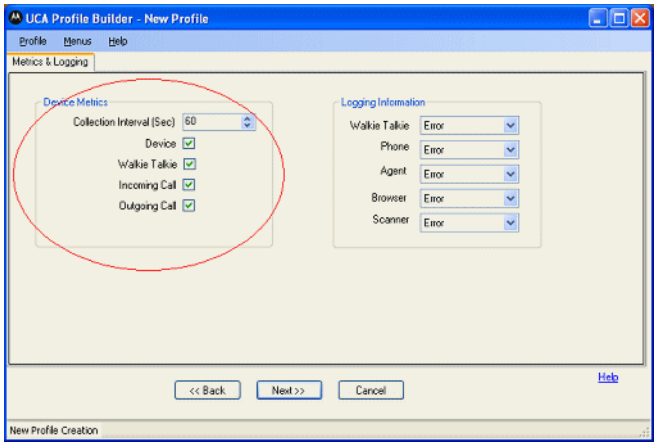


Figure 8-15 *Metrics*

Table 8-5 *Metrics Settings*

Field	Description
Collection Interval	Specifies the send interval for sending the metrics data to the server. This applies only to device and push to talk messages.
Device	Check this box to include device specific information as part of metrics data sent to the server.
Walkie Talkie	Check this box to include Walkie-Talkie messages as part of metrics data sent to the server.
Incoming Call	Check this box to include incoming call metrics as part of metrics data sent to the server.
Outgoing Call	Check this box to include outgoing call metrics as part of metrics data sent to the server.

Logging

The Logging section determines the levels of logging for various device functions such as *Walkie-Talkie, Phone*, *Agent, Browser* and Scanner, or whether to turn logging off for each of these.

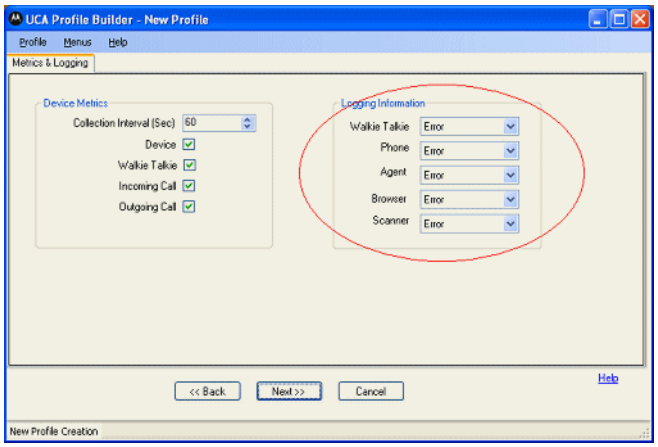


Figure 8-16 *Logging*

Table 8-6 *Logging Settings*

Field	Description
Off	No logging.
Info	Logs high level information such as device and server IP addresses.
Error	Logs all error messages along with information messages.
Warn	Logs information, error and warning messages.
Debug	Logs all messages

Device Settings

This section captures the device settings such as backlight and RF sample time.

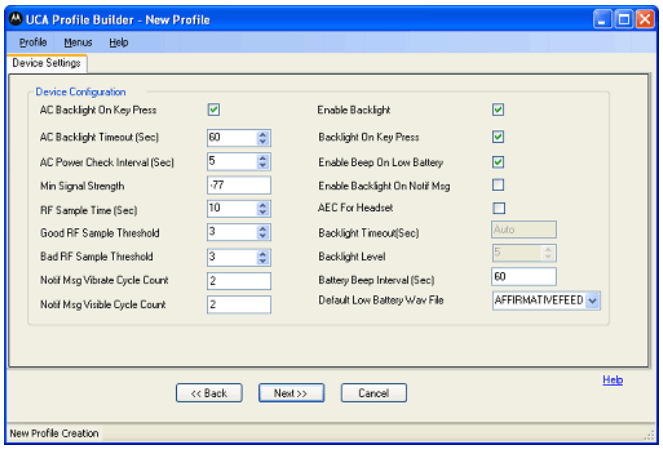


Figure 8-17 *Device Settings*

Table 8-7 *Device Settings*

Field	Description
AC Backlight On Key Press	Press key to turn on the backlight (device is in the cradle).
AC Backlight Timeout	Sets the time in seconds that the backlight stays on (device is in the cradle).
AC Power Check Interval	Sets the time in seconds that the device uses to check if it is in the cradle.
Min Signal Strength	Sets the minimum RF signal strength to be connected (the numbers are negative; the larger the negative number, the weaker the signal, e.g., -70 is weaker than -60).
RF SampleTime	Sets the time in seconds for the RF signal strength sampling.
Good RF SampleThreshold	Sets the number of times the RF sample signal strength must be above the minimum signal strength before the bad sample count is reset.
Bad RF SampleThreshold	Sets the number of times the RF sample signal strength must be below the minimum signal strength before the Network Connection Lost message displays.
Notif Msg Vibrate Cycle Count	The number of times the device should vibrate on receipt of notification message.

Table 8-7 *Device Settings (Continued)*

Field	Description
Notif Msg Visible Cycle Count	The number of times the device should show visible indication on receipt of notification message.
Enable Backlight	Check the check box to enable the device backlight on the device. Uncheck the check box to disable the backlight on the device (when on battery power) regardless of the values of other parameters such as *Enable Backlight On Key Press*.
Backlight On Key Press	Check the check box to turn the device backlight on when a key is pressed (on battery power).
Enable Beep On Low Battery	Check the check box to enable the beep when the battery is low function.
Enable Backlight On Notif Msg	Check the check box to enable the backlight when a notification is delivered.
AEC For Headset	Check this box to enable echo cancellation for the headset mode. By default echo cancellation is disabled. This parameter is applied only to the headset mode and it is enabled for the handset at all the times.
Battery Beep Interval	Sets the time interval (seconds) between beeps when the battery is low.
Default Low Battery Wav File	Sound file that plays when the battery goes below the threshold level.

Client Settings

Use the *Client Settings* to set the clock and configure other operational parameters (such as timeouts).

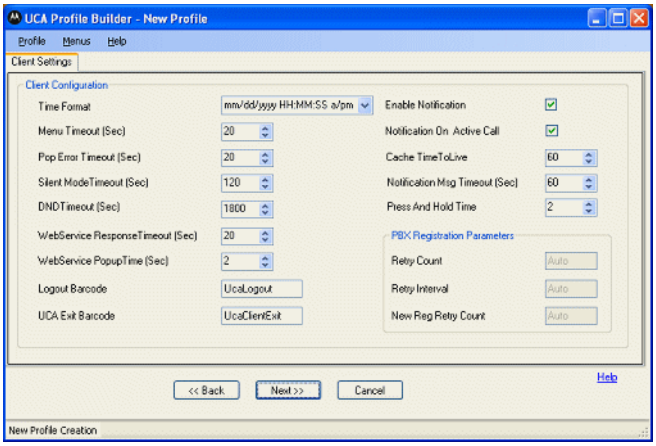


Figure 8-18 *Client Settings*

Table 8-8 *Client Settings*

Field	Description
Time Format	Select time format from the drop-down list (the time format displayed on the device).
Menu Timeout	The amount of time in seconds that a menu displays (without a key press).
Pop Error Timeout	The amount of time in seconds an error message displays.
Silent Mode Timeout	The amount of time in seconds that the UCA client application mutes after pressing Mute.

Table 8-8 *Client Settings (Continued)*

Field	Description
DND Timeout	The amount of time in seconds that the UCA client application stays in *Do not disturb* mode.
WebService Response Timeout	The amount of time a Web service call stays connected waiting for a response.
WebService PopupTime	The number of seconds to wait for a response from the server before displaying a *Waiting for Server* message.
Logout Barcode	Set the *Logout Barcode* value to allow the user to scan a bar code to logout. Set this value to "" to disable log off by bar code scanning. Note that, when enabled, scanning a bar code to log off works only when the user is logged in.
UCA Exit Barcode	Set the *UCA Exit Barcode* value to allow the user to scan a bar code to exit the UCA client application. Set this value to "" to disable exiting the UCA client application by bar code scanning. Note that, when enabled, scanning a bar code to exit the UCA client application works only when the user is logged in.
Enable Notification	Checking this enables notifications, unchecking this disables notifications.
Notification On Active Call	Check to enable notifications while on a phone call, uncheck to disable.
Cache TimeToLive	Ignored, currently not implemented.
Notification Msg Timeout	The time in seconds for a notification message to display.
Press And Hold Time	Ignored, currently not implemented.

Alert Settings

Use the Alert Settings section to configure various alerts on the device for various events. It defines the type of indication to provide in terms of the following:

- Tone played
- Visible indicator (lighting up LED in different colors and/or blinking the LEDs)
- Vibrate Indicator.

For example: The *NotifCallActive* setting controls the kind of alert provided when a notification is sent (during an active call). This defines the kind of tone to play, whether the LED blinks or not, which LED blinks and how fast. This also controls if the device vibrates and how fast.

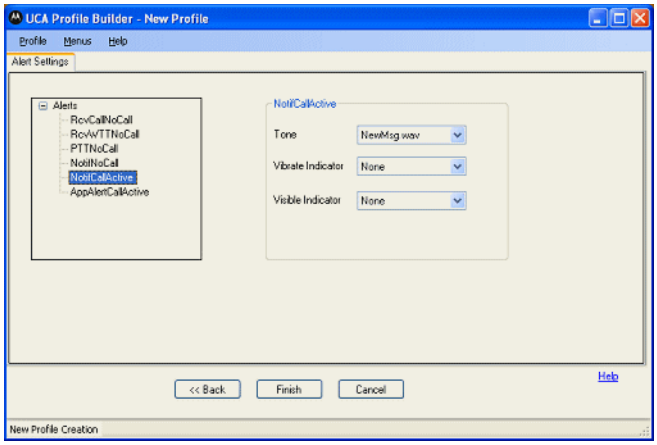


Figure 8-19 *Alert Settings*

Choose the setting and select the options in the corresponding alert details section of the window. Click *Help* for detailed descriptions of each alert type.

RcvCallNoCall:
The setting defines the alert behavior when a phone call is received. These attributes are applied as-is, for visible and vibrate indicators. The tone is selected from the database; the value cannot be set for this attribute.

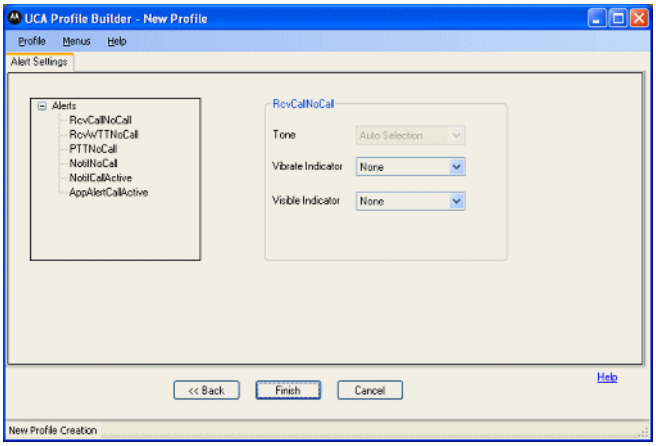


Figure 8-20 *RcvCallNoCall*

RcvWTTNoCall:
The setting defines the alert behavior when receiving Walkie-Talkie audio data (while not on an active call). The only applicable attribute is tone. The tone attribute applies as is.

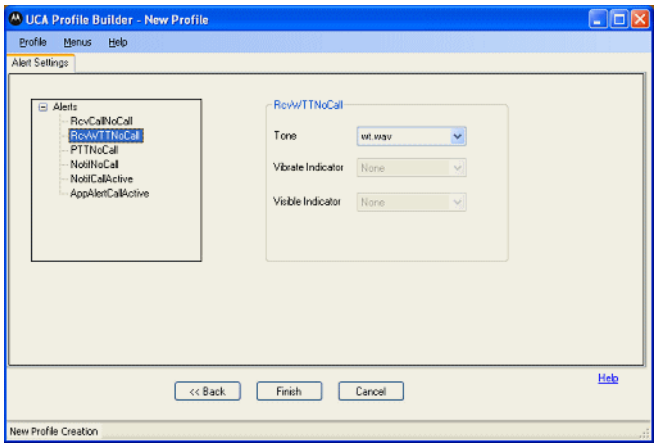


Figure 8-21 *RcvWTTNoCall*

PTTNoCall:
The setting defines the alert behavior when either **Walkie-Talkie** key is pressed when not on an active call. The only applicable attribute is tone.

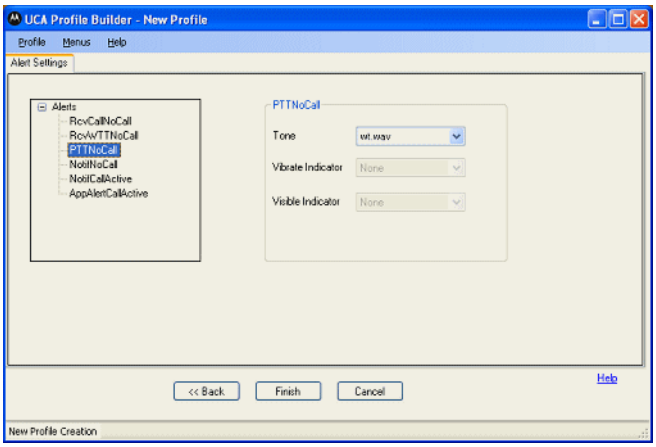


Figure 8-22 *PTTNoCall*

NotifNoCall:
The setting defines the alert behavior when a notification message is received (not on a phone call).

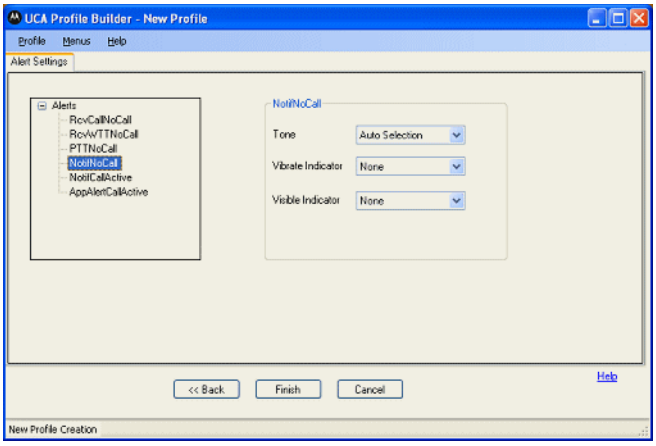


Figure 8-23 *NotifNoCall*

There are three alert resources:

1. Tone: A tone plays based on information from either the data within the notification message or from the profile data.

   The device checks if the notification message has data to play the tone and plays the tone if the data is available.

   If the notification message has no data for tone, the user profile is checked for the information. If the profile contains valid tone attribute information (associated with *NotifNoCall*) the tone plays.

   The information in the notification takes precedence over the profile.

2. LED: The LED flashes based on information from either the data within the notification message or from the user profile.

   The device checks if the notification message has the visible indicator data. If data is available, the LED flashes.

   If the notification message has no data for the visible indicator, the user profile is checked for the information. If the profile contains the visible indicator information (associated with *NotifNoCall*) the corresponding LED flashes.

   The information in the notification takes precedence over the profile.

3. Vibrate: The device vibrates based on information from either the data within the notification message or from the user profile.

   The device checks if the notification message has the vibrate indicator. If data is available, the device vibrates.

   If the notification message has no data for the vibrate, the user profile is checked for the information. If the profile contains the vibrate information (associated with *NotifNoCall*) the device vibrates.

   The information in the notification takes precedence over the profile.

   NotifCallActive:
   The setting defines the alert behavior when receiving a notification message (on an active phone call).

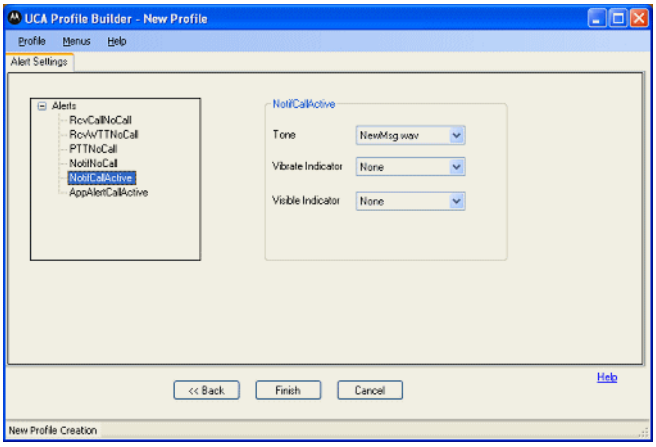


Figure 8-24 *NotifCallActive*

The behavior of the alerts is the same as *NotiNoCall*, except that the settings in the profile take precedence over the message settings. These alerts function only if the *NotifAllowOnCallActive* is set to *Yes*.This setting is under the device configuration (see device configuration *page 7-7*).

AppAlertCallActive:
The setting defines the receiving a notification alert behavior when the device is already running an application and an active call is in progress. The allowed values for tone, visible and vibrate indicators are boolean values,

whether the alerts are allowed or not in that state. If the tone, visible and/or vibrate indicator is allowed, the behavior of these alerts follows the same rules as that of *NotifCallActive*.

The device checks if the alerts attributes (tone, visible, and vibrate) are enabled.

If the alert attributes are enabled, the attribute behavior applicable for *NotifCallActive* will apply.

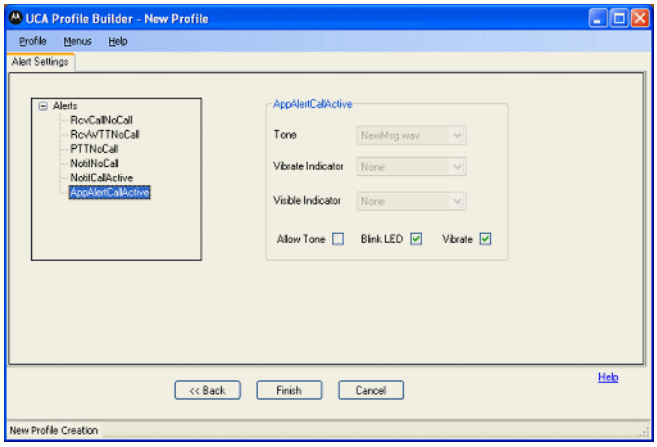


Figure 8-25 *AppAlertCallActive*

AppAlertCallActive alert has three configurable options:

- *Allow Tone* - Check box to activate tone
- *Blink LED* - Check box to display visible indicator
- *Vibrate* - Check box activate vibrate indicator

NOTE Attributes defined for *NotifNoCall, NotifCallActive*, and *AppAlertCallActive* apply only when the client setting *Enable Notification* is set to *Yes*. If this attribute is set to *No*, the notification is not sent to the device and none of the attributes defined for notification will apply.

Create Menus

Click *Help* for detailed descriptions of menus and menu states.

To create the main menu:

1. Run the *Profile Builder* application and select *Menus* > *New*. The *Menus & Menu States* window appears.

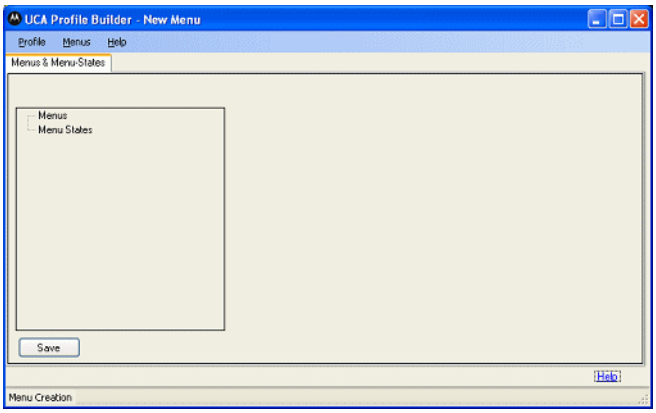


Figure 8-26 *Menus & Menu States*

2. Right-click *Menus* and select *Add Menu*.

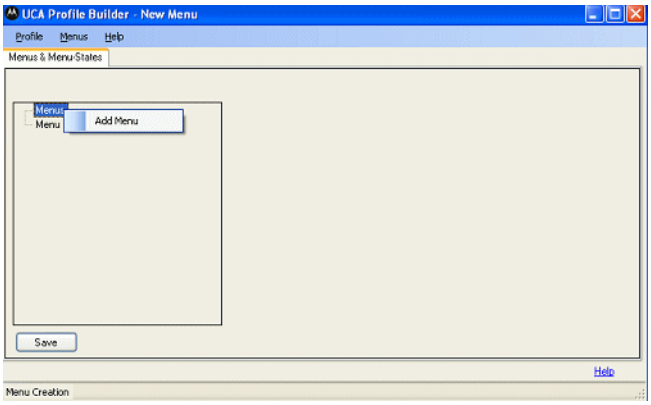


Figure 8-27 *Add Menu*

3. Click the plus sign to the left of menus to expand the tree.

4. Select the newly added menu item under *Menus* (*Menu11*) to display *Main Menu*.

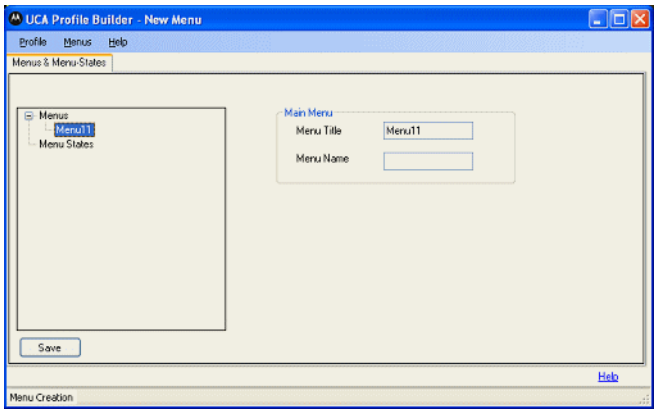


Figure 8-28 *New Added Menu*

5. Type a menu title (e.g., *Options*) in the text box under *Main Menu*.

6. Type a menu name (e.g., *AppCallActive*) in the text box under *Main Menu*.

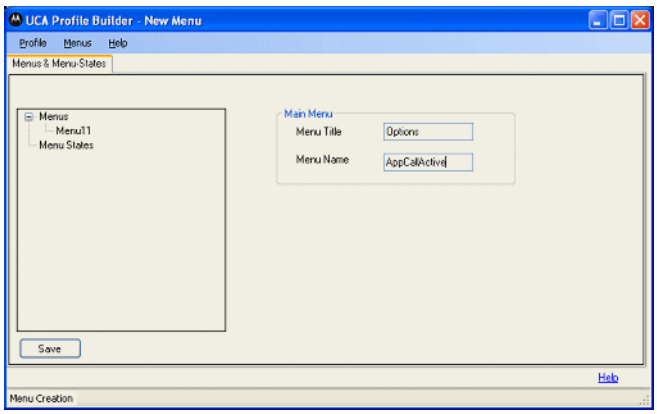


Figure 8-29 *Enter Menu Name*

Add Menu Elements

There are four types of menu elements:

1. Static menu element: Refers to one more static menu in the profile.

2. Dynamic menu element: This is a menu generated dynamically either in the device itself or from external source through a web request.

3. Client command: Refers to a command executed when a menu item is selected.

4. Server action: Refers to a special action taken by executing the URI specified on the UCA server.

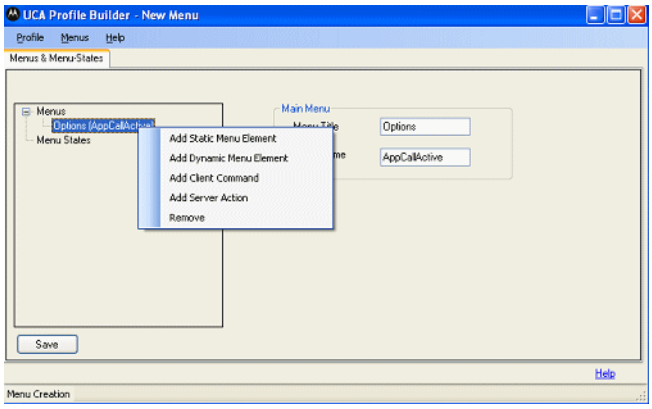


Figure 8-30 *Menu Elements*

Add a Static Menu Element

To delete any menu, select the menu to delete, right-click and select **Remove**.

1. Right-click the main menu - Options (AppCallActive) - and select *Add Static Menu Element*.
2. Select the new *Static* menu item (*New Menu2*).

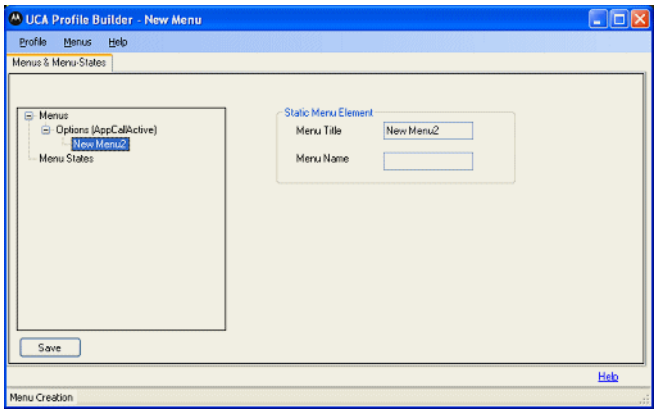


Figure 8-31 *Add a Static Menu Element*

3. Enter a *Menu Title* (e.g. Transfer) in the text box under *Static Menu Element*.

4. Enter a *Menu Name* (e.g., TransferPhoneBook) in the text box under *Static Menu Element*.

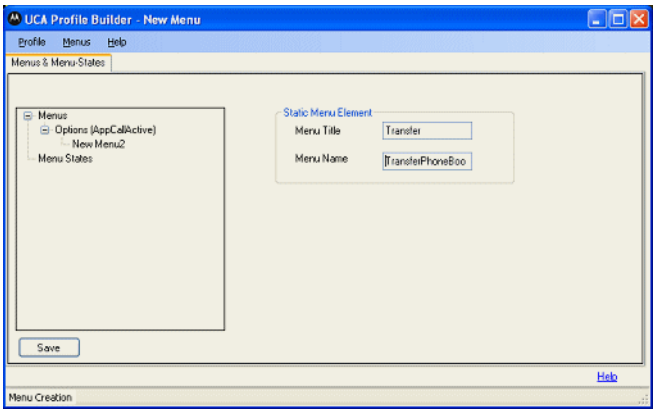


Figure 8-32 *Enter Menu Name*

Add a Dynamic Menu Element

1. Right-click the main menu - *Options* (*AppCallActive*) - and select *Add Dynamic Menu Element*.
2. Select the new dynamic menu item (*Dynamic Menu3*).

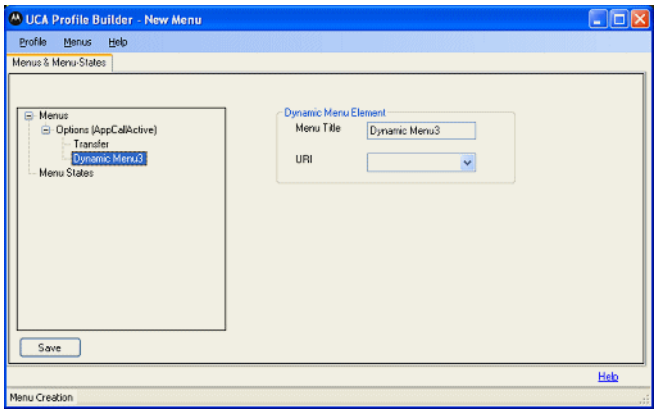


Figure 8-33 *Add a Dynamic Menu Element*

3. Enter a menu title (e.g., *FirstName*) in the text box under *Dynamic Menu Element*.
4. Select a URI (e.g., *First Name URI*) from the *URI* drop-down list under *Dynamic Menu Element*.

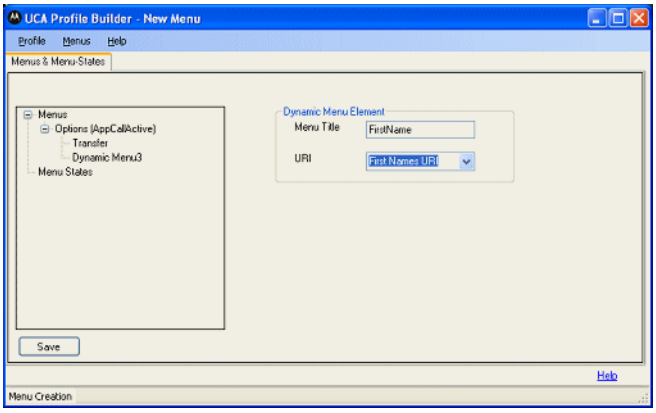


Figure 8-34 *First Name URI*

Add a Client Command Menu Element

1. Right-click the main menu - *Options* (*AppCallActive*) - and select *Add Client Command*.
2. Select the new *Client Command* item (*Client Command4*).

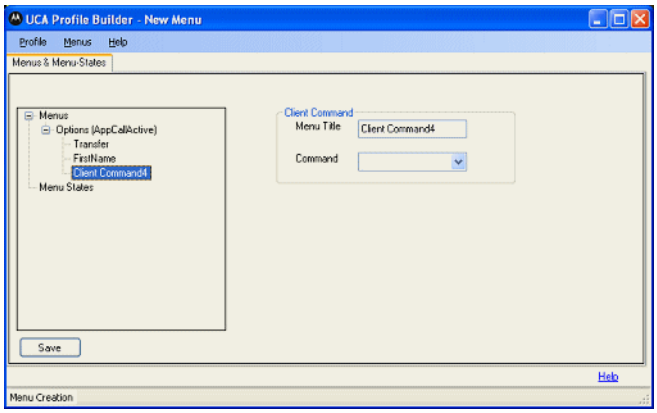


Figure 8-35 *Add a Client Command Menu Element*

3. Enter a *Menu Title* (e.g., *About*) in the text box under *Client Command*.
4. Select a *Client Command* (e.g., *About*) from the *Command* drop-down list under *Client Command*.

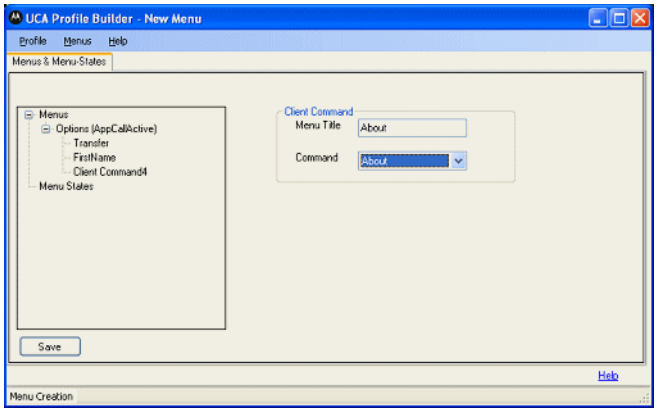


Figure 8-36 *Client Command*

Add a Server Action Menu Element

1. Right-click the main menu - *Options* (*AppCallActive*) - and select *Add Server Action* from the options listed.
2. Select the new *Server Action* item (*Server Action5*).

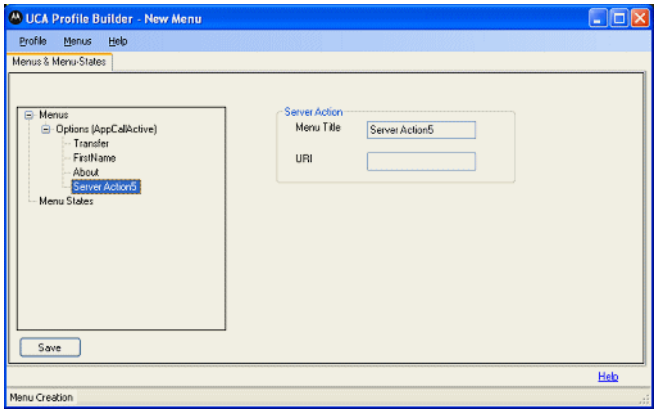


Figure 8-37 *Add a Server Action Menu Element*

3. Type a *Menu Title* (e.g., Price Check) in the text box under *Server Action*.
4. Type a valid *URI* in the text box under *Server Action*.

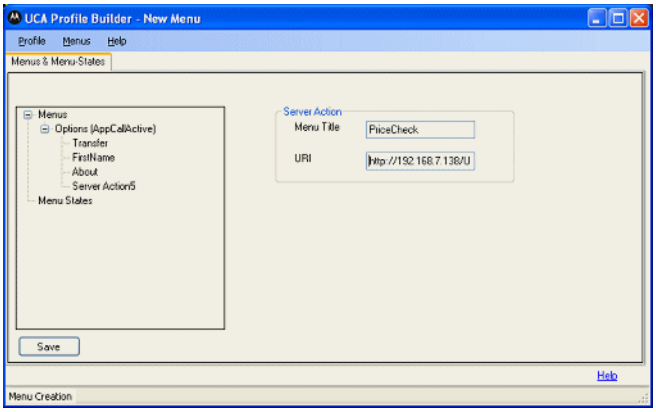


Figure 8-38 *Enter Menu Title*

5. Click **Save** and proceed to *Add Menu States*, when finished adding menus.

NOTE Use the sample profile provided under *C:\Program files\Motorola\UCA Server\Profiles* and modify it to suit requirements rather than creating a new menu.

Add Menu States

Menu states define the operational state of the UCA client and the left and right soft key actions in that state. The actions can be either a reference to a menu or an execution of a command.

There are eight menu states:

1. *AppCallActive*: Call is active and an application is running.
2. *AppNoCall*: Application is running but no active call.
3. *AppQuietMode*: Application is running, device is in a DND (*Do not disturb*) mode.
4. *AppCallOnHold*: Application is running, and the call is on hold.
5. *DesktopCallActive*: Call is active and is engaged.
6. *DesktopNoCall*: In *Desktop* mode, not engaged in an active call and no application is running.
7. *DesktopQuietMode*: In DND mode and no application is running.
8. *DesktopCallOnHold*: Call is on hold.

The menu state defines what is associated with the left soft key and right soft key. The soft keys can be associated with menus or commands.

Add Menu States

1. Run the UCA *Profile Builder* application and select *Menus* > *New*. The *Menus & Menu States* window displays.

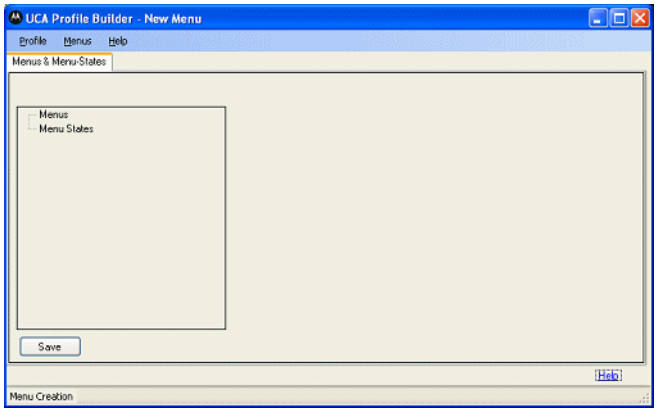


Figure 8-39 *Add Menu States, Menus & Menu States*

2. Highlight *Menu States*.
3. Right click *Menu States* and select *Add Menu States* > *AppCallActive*.

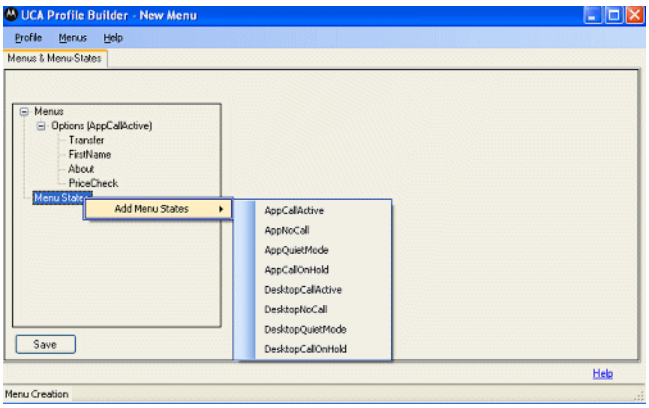


Figure 8-40 *Menu States*

4. Click the plus sign to the left of *Menu States* to expand the tree.
5. Select the newly added menu item under *Menu States* (*AppCallActive*) to display the details for the menu state.

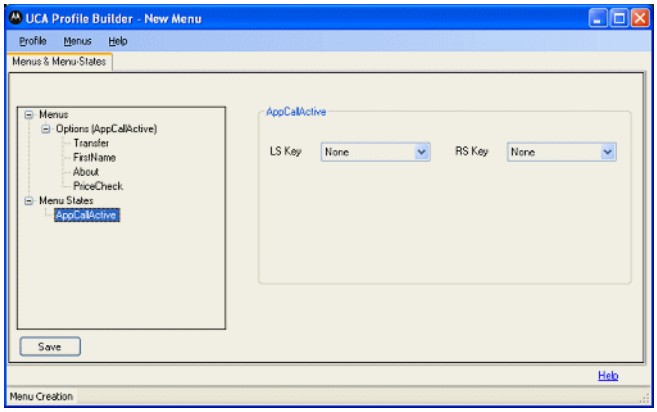


Figure 8-41 *Menu States Details*

6. Select an *LS Key* (left soft key) and/or *RS Key* (right soft key) operative item from the drop-down lists.

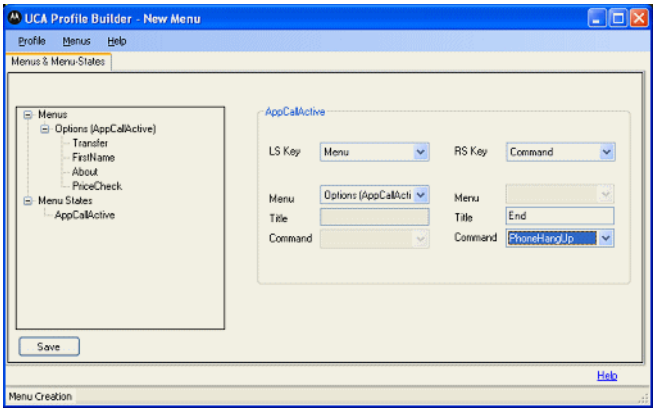


Figure 8-42 *Select Soft Key*

7. For each key select a menu item from the drop-down list (if applicable).

   The menu items drop-down list includes the previously added main menus.
8. For each key, if applicable, select a *Command* and enter a *Title*.
9. Repeat step 3 through step 8, as needed, to add the required menu states (*AppCallActive*, *AppNoCall*, *AppQuietMode*, *AppCallOnHold*, *DesktopCallActive*, *DesktopNoCall, DesktopQuietMode* and *DesktopCallOnHold*).
10. Save the XML file.

Chapter 9 UCA Server WebConsole Graphical User Interface

Overview

This chapter provides information about the UCA server WebConsole GUI and how to use it for UCA server configuration and setup.

Valuable topics in this chapter include:

- UCA Server WebConsole Interface Overview (*page 9-2*)
- Login to the UCA server system (*page 9-3*)
- System Status tab view (*page 9-5*)
- Phone and Walkie-talkie tab view (*page 9-7*)
- Phone extension import file format (*page 9-13*)
- Deleting phone extensions (*page 9-17*)
- Device User Accounts tab view (*page 9-18*)
- Deleting a user (*page 9-24*)
- Profile Management tab view (*page 9-26*)
- Create an associated base profile (*page 9-27*)
- Edit an associated base profile (*page 9-28*)
- Delete an associated base profile (*page 9-29*)
- Roles and Groups tab view (*page 9-30*)
- System Management tab view (*page 9-37*)
- UCA server WevConsole GUI utilities (*page 9-52*)
- Import utility (*page 9-55*).

UCA Server WebConsole Interface Overview

The UCA server WebConsole Interface is a Graphical User Interface (GUI) for UCA server configuration and setup. The GUI allows an administrator to view, create, edit and delete settings and entries in the UCA server.

The interface is segmented into the following sections:

- System Status
- Device User Accounts
- Phone and Walkie-Talkie
- Profile Management
- Roles and Groups
- System Management.

NOTE 1. For a complete checklist outlining all the development and deployment activities required to implement a CA50 solution, see the *Development and Deployment Checklist on page 2-1*.

2. For a table listing the most commonly requested topics within this manual, see the Quick Startup Guide on the last page of the manual.

Login to the UCA Server System

To login to the UCA server system:

1. Ensure the UCA server installation was completed successfully. For more information see *Validate Successful Server Installation and Proper Performance on page 6-45*.

2. Launch the UCA server WebConsole by double clicking the *UCA server Web Console* icon on the server desktop.
   or
   Launch the IE browser and point the browser to one of the following URLs:

   - When logging in to the UCA server software locally, software is loaded on the computer being used and the following URL is applicable:
     http://localhost/Motorola/UCA/UCAConsole/Login.aspx.

   or

   - For remote login to the UCA server software, use the following URL and substitute the server IP address into the portion of the *<server IP address>* URL shown below. For example replace *<server IP address>* with *192.168.0.253* (use your relevant server IP address).
     http://*<server IP address>*/Motorola/UCA/UCAConsole/Login.aspx.

3. The following window displays.

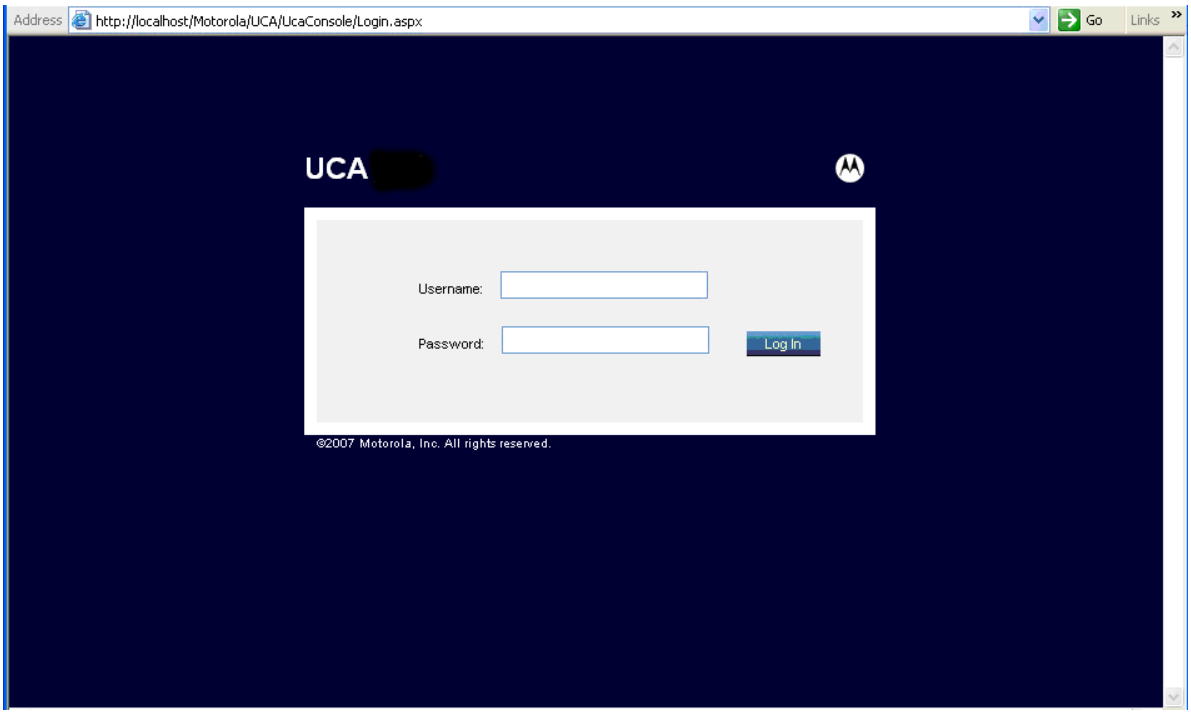


Figure 9-1 *WebConsole GUI Login*

4. Enter the *Username* and *Password* set for you by the UCA Administrator.

IMPORTANT At the initial login to the Web Console (for details see *page 9-3*), it is required to use the default Web Console administrator *Username* and *Password*:

- *Factory Default Web Console Username*: *admin*
- *Factory Default Web Console Password*: *admin*

It is strongly suggested that the default administrator–*Password* be changed after the initial login. See *Editing a Console User Password on page 9-46* for more information.

Note that the *Username* and *Password* entered during the UCA server software installation process, (*step on page 6-13*) are for SQL database access and should not be mistaken with the default Web Console administrator Username and Password shown above.

5. Click **Log In**.

6. The *System Status* tab displays showing the *Server Status* view.

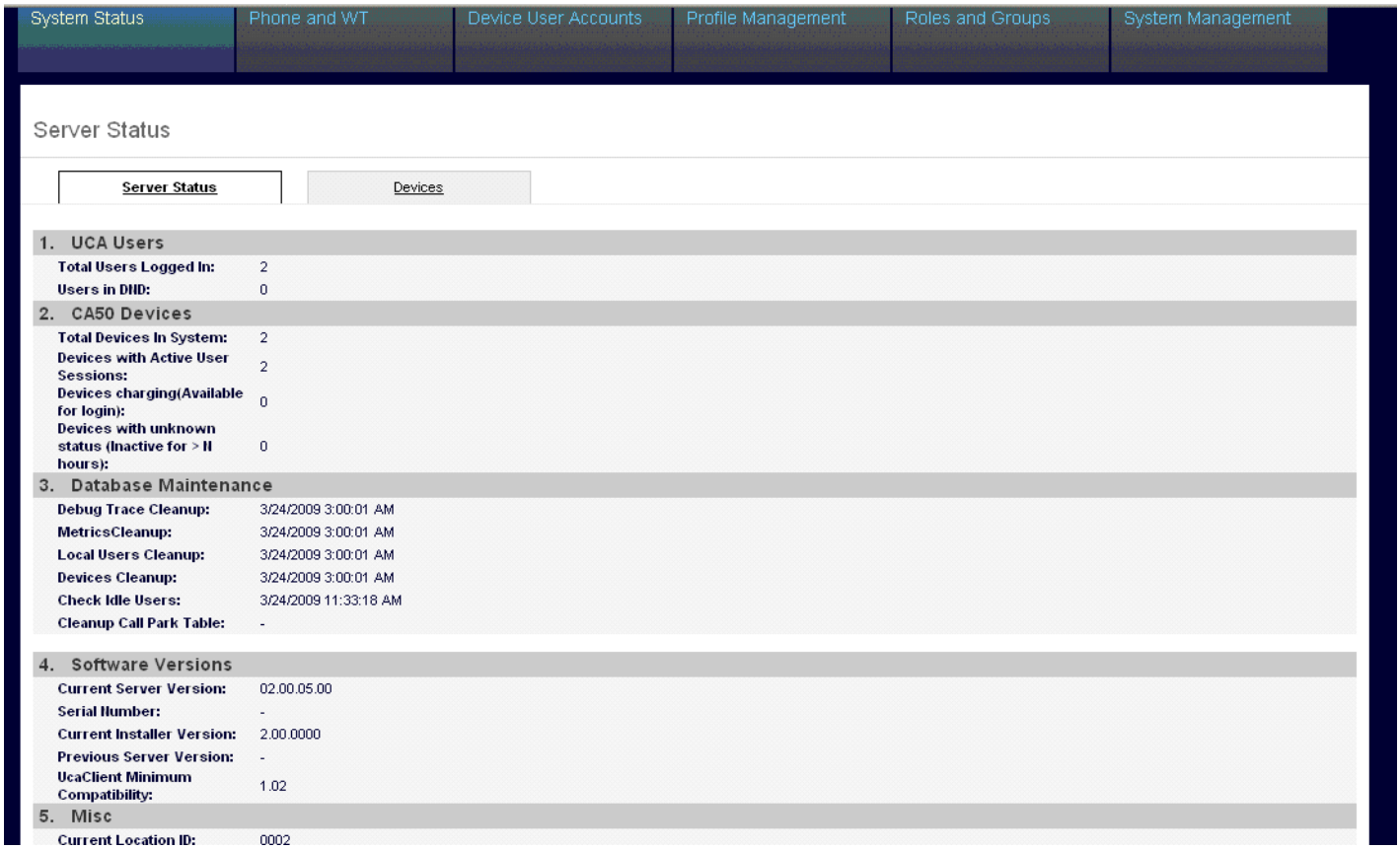


Figure 9-2 *System Status Tab*

System Status Tab View

Click the *System Status* tab on the UCA server WebConsole GUI window to display the view.

The *System Status* tab has two views:

- *Server Status* (*Figure 9-3*)
- *Devices* (*Figure 9-4*).

To toggle between views, click on a tab: **Server Status** and/or **Devices**.

Server Status

Server Status view displays the status of all UCA Users, CA50 Devices, Database Maintenance, Software Versions and Miscellaneous entries.

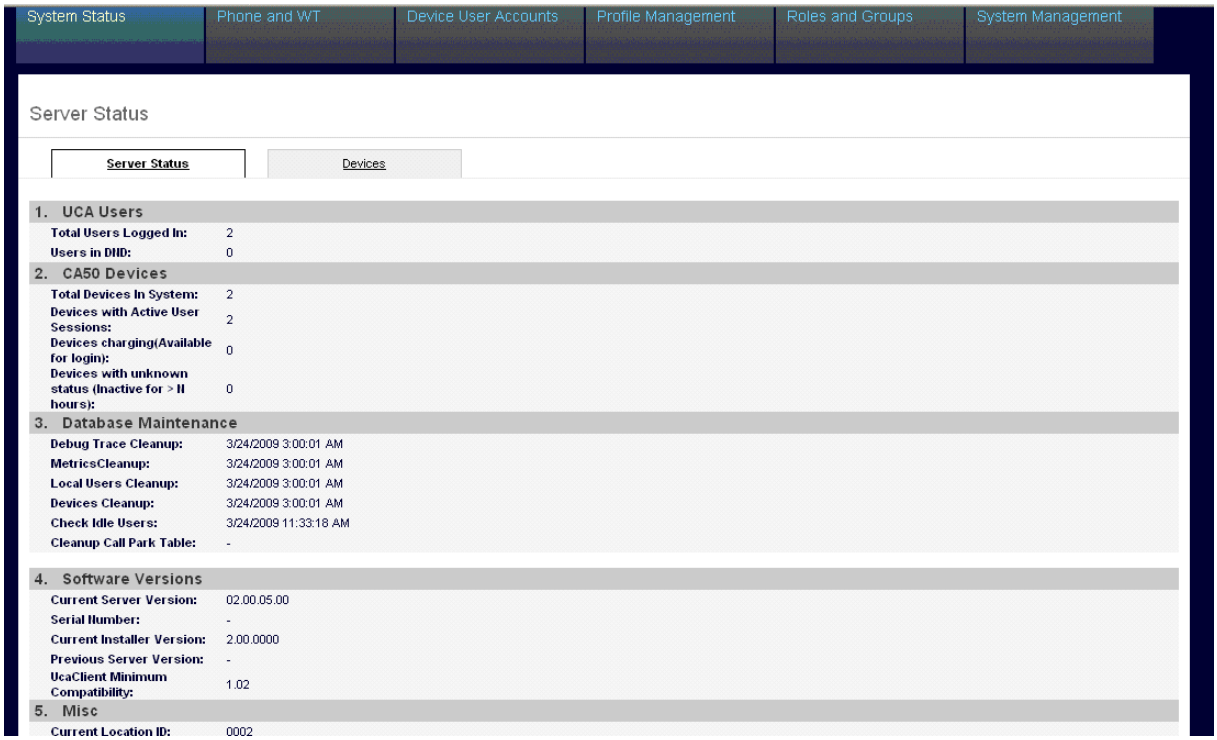


Figure 9-3 *Server Status View*

Devices

Devices view displays a list of devices recently seen by the UCA server. Displayed information includes: Device ID, Serial Number, Device model, Last Known User ID,– Last Known User Name, User Status, Assigned Phone Extension, Last Login and Activity Time.

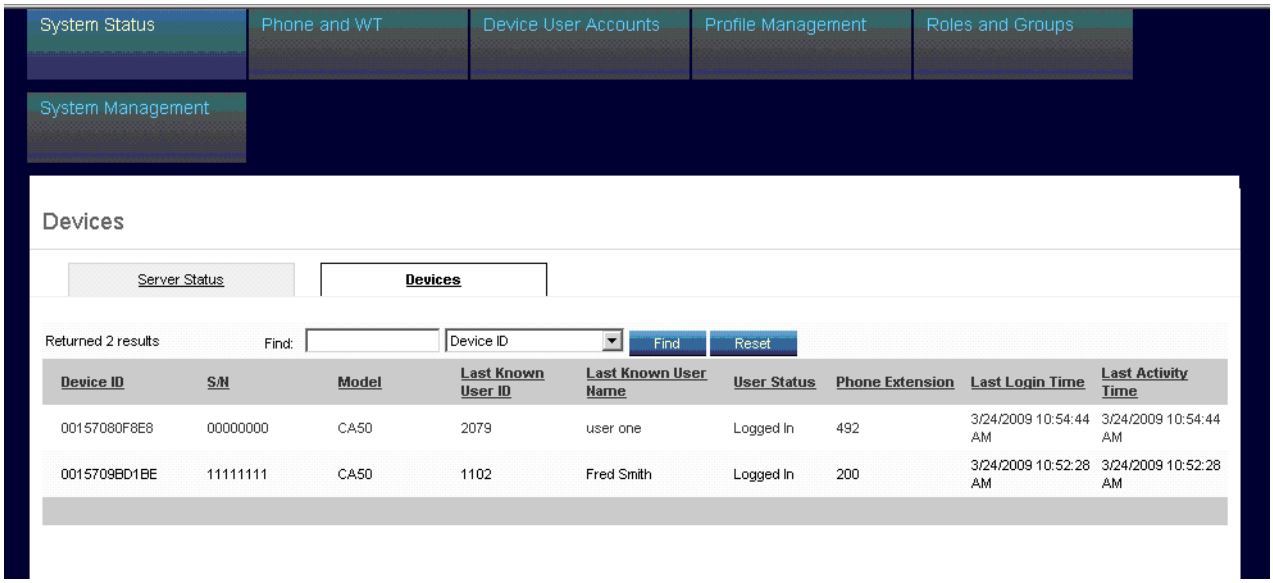


Figure 9-4 *Devices View*

Phone and Walkie-Talkie Tab View

Click the *Phone and Walkie-Talkie* tab on the UCA server WebConsole GUI window to display the view.

The *Phone and Walkie-Talkie* tab has three views:

- *Phone Extension Status* (*Figure 9-5*)
- *Phone Extension Pool* (*Figure 9-6*)
- *Additional Phonebook Entries* (*Figure 9-14*).

Each view contains all information (phone extensions) about users defined in the UCA server. In these views create, edit, and delete phone extensions for CA50 and other phone devices. To toggle between views, click on a tab: **Phone Extension Status**, **Phone Extension Pool** and/or **Additional Phonebook Entries**.

Phone Extension Status

The *Phone Extension Status* view displays the status of all phone extensions that can be assigned to a CA50. It also lists any phone extensions that failed when attempting to register with the PBX. This view shows *Extension, Group, Extension Status, Current Owner External ID* and *Current Owner IP Address* of all active users in the UCA server.

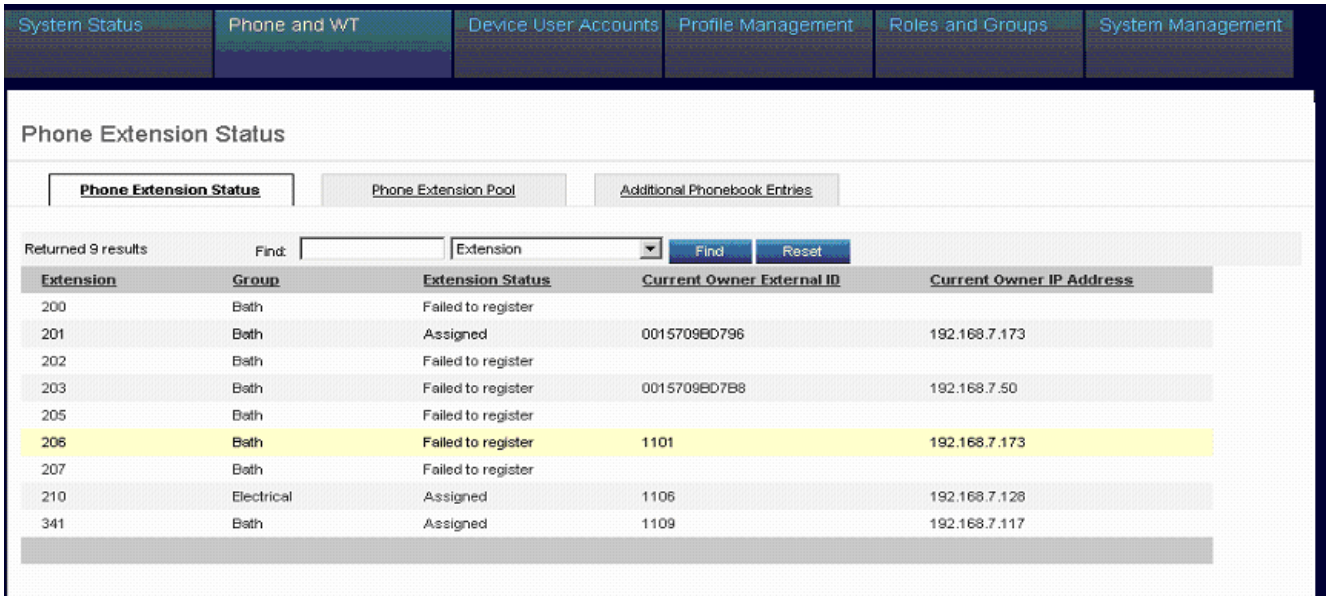


Figure 9-5 *Phone Extension Status View*

Phone Extension Pool

The *Phone Extension Pool* view displays the pool of extensions defined in the UCA server that are assignable to a CA50. The view includes: phone *Extension, Group, Static Owner External ID*, *Extension Type* and *Phonebook Description* of all phone extensions in the UCA server. For SCCP protocol, the phone extension pool view see *Figure 9-6*. Red cross indicates that extension is non-SCCP see *Figure 9-7*.

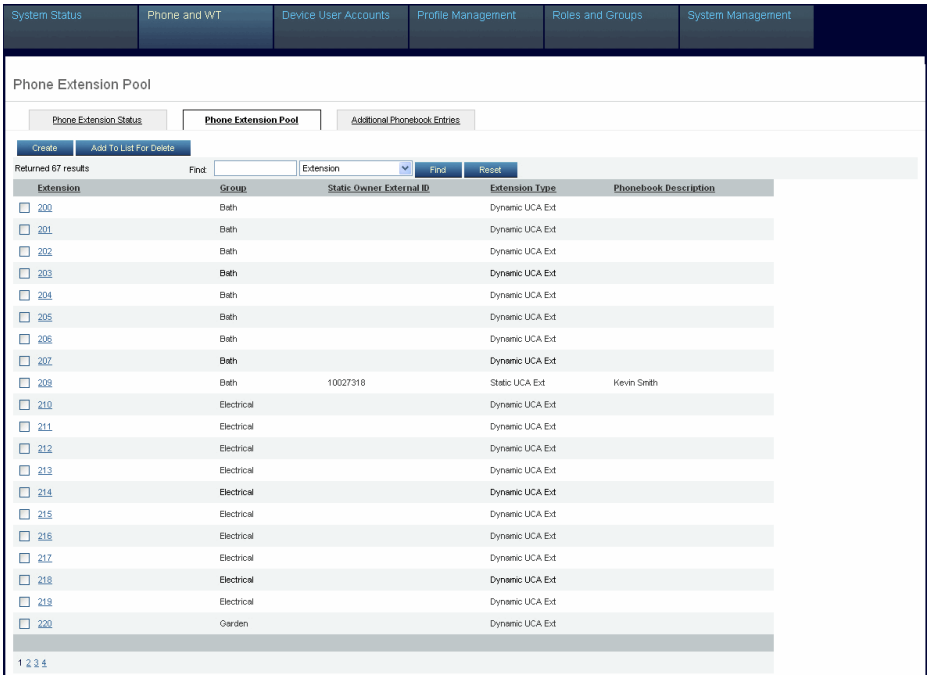


Figure 9-6 *Phone Extension Pool View.*

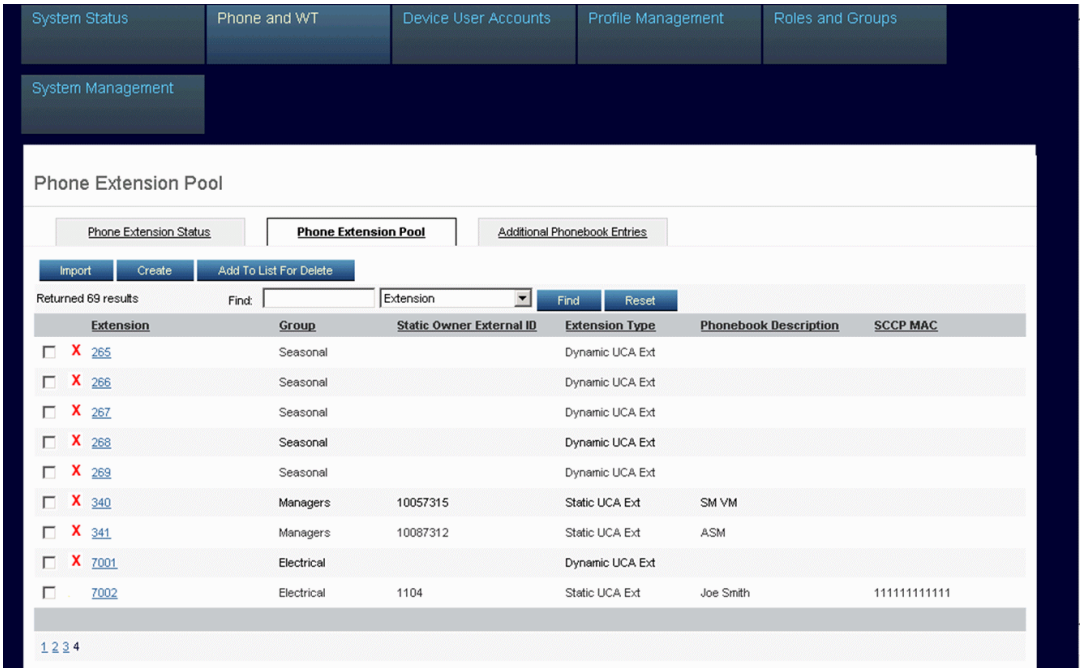


Figure 9-7 *Phone Extension Pool View (non-SCCP)*

Creating a UCA Phone Extension

To create a phone extension in the UCA server:

1. Click the *Phone and Walkie-Talkie* tab > **Phone Extension Pool** tab > **Create**.
   The *Create Phone Extension* window displays.

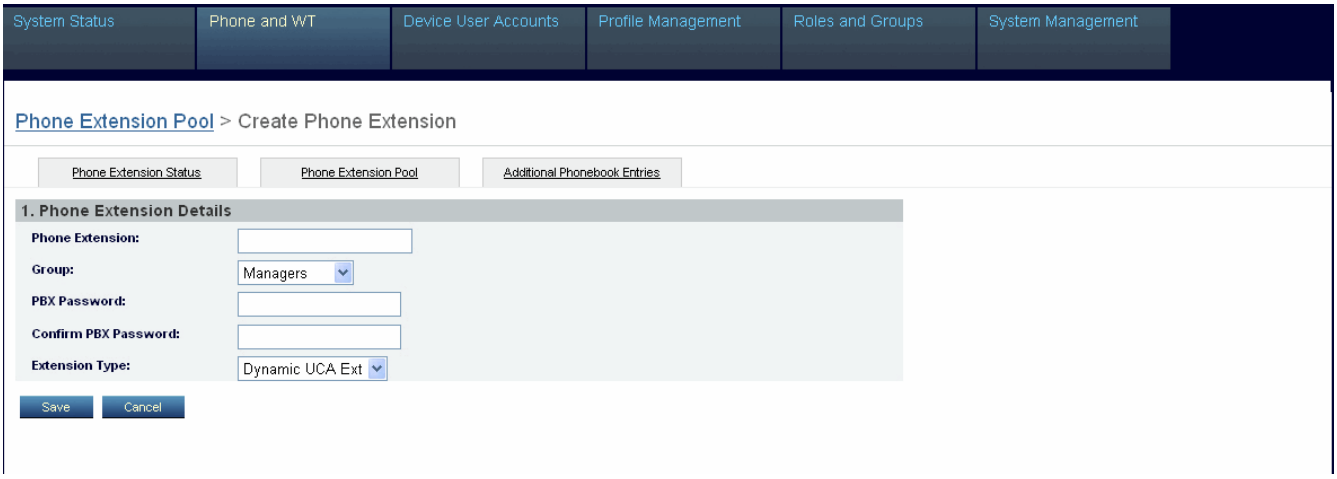


Figure 9-8 *Create Phone Extension Window*

2. For SCCP, (only) an additional parameter is required, the SCCP MAC address. This mandatory field is applicable for SCCP only. The Mac Address is a unique identifier that provides the mappings between an extension and a Mac Address in the PBX system. The Mac Address mentioned in the console is the same address that is entered in the PBX table corresponding to the extension.

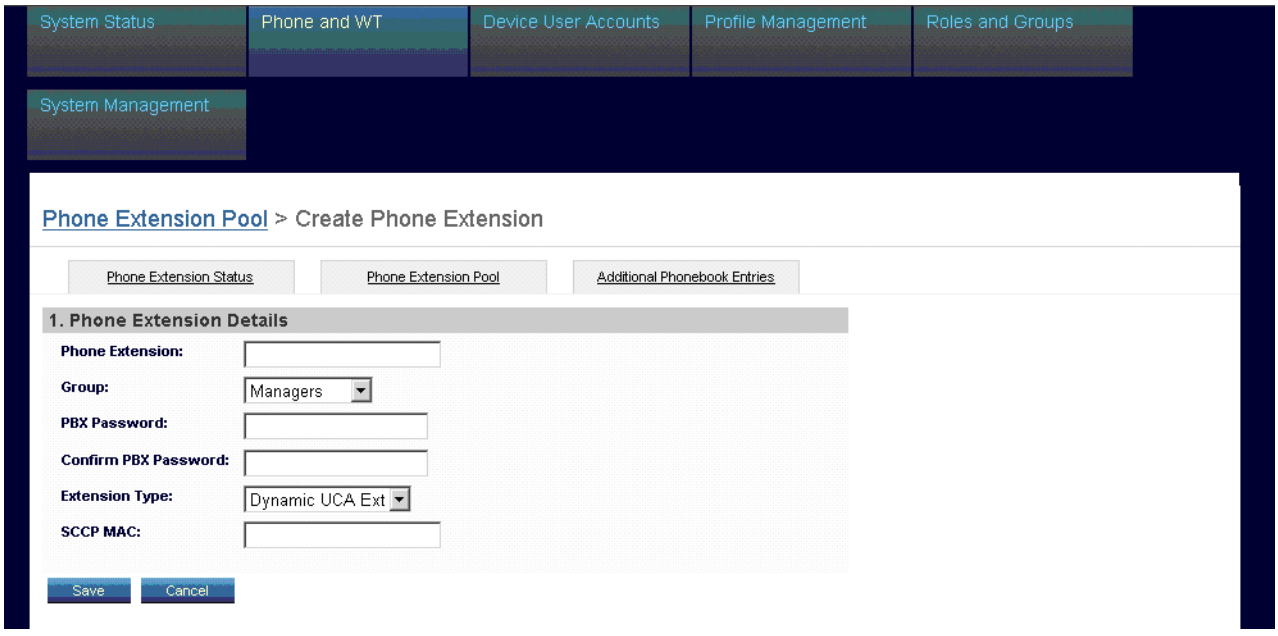


Figure 9-9 *Create Phone Extension Window (SCCP)*

3. Enter a *Phone Extension* and *PBX Password* and in the appropriate text boxes. The PBX password option is at the discretion of the UCA server Administrator.

NOTE The PBX password depends on PBX configuration. Some PBXs are configured to require a password during the phone extension assignment process. Use this field to supply the PBX password in such cases.

The *Static Owner External ID*, *Phonebook Description* and *Always Show* properties only apply to Static UCA Extension types (see *Figure 9-12*). When the extension type is static the phone extension only gets handed out when a user logs in with a matching *External ID*. The *Always Show* property determines if this extension always gets displayed in the phone book regardless of whether the assigned user is logged in. This might be used if the user has voice mail or a call forwarding feature. The *Phonebook* description then displays in the phone book when the user is not logged in.

4. Select a *Group* from the drop down list menu.
5. Select an *Extension Type* from the drop down list menu.
6. Click **Save** to store the user or **Cancel** to terminate the session.

Editing the Properties Connected to a UCA Phone Extension

To edit the properties connected to a UCA phone extension in the UCA server:

1. Click the *Phone and Walkie-Talkie* tab > **Phone Extension Pool** tab.
2. Click the phone extension to edit from the list of UCA extensions in the *Extension* column (see *Figure 9-6 on page 9-8*). The *Phone Extension Detail* window displays.

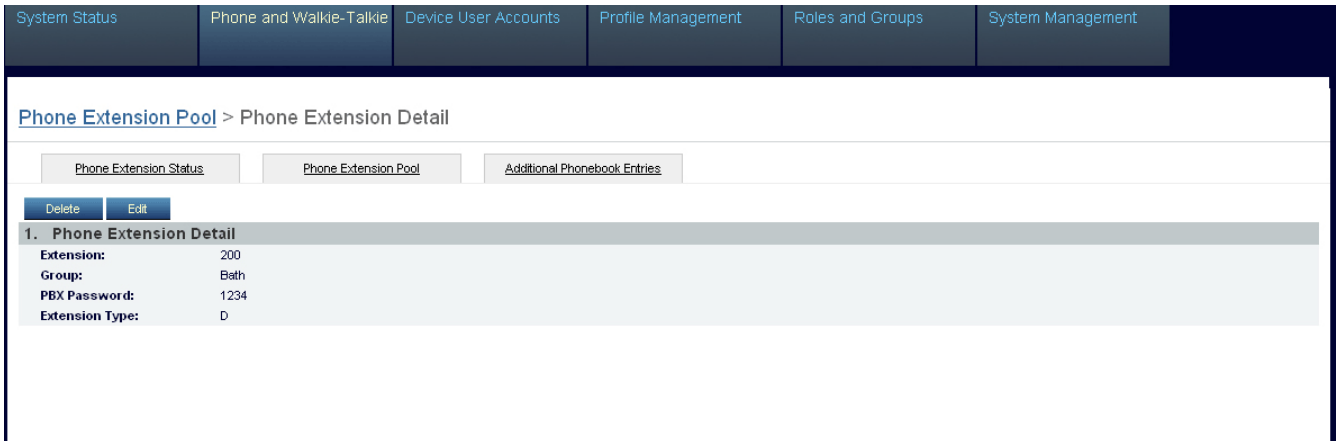


Figure 9-10 *Phone Extension Detail Window*

3. *For SCCP the SCCP Phone Extension Detail window displays.*

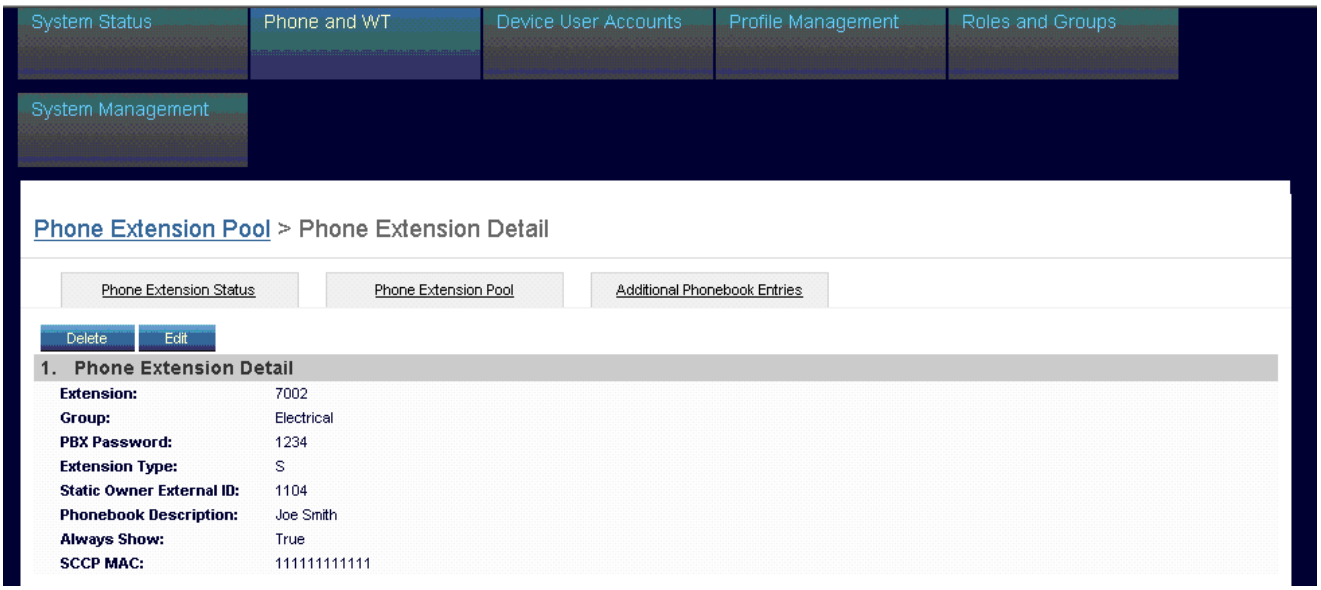


Figure 9-11 *SCCP Phone Extension Detail Window*

4. Click **Edit**.
   The *Edit Phone Extension* window displays.

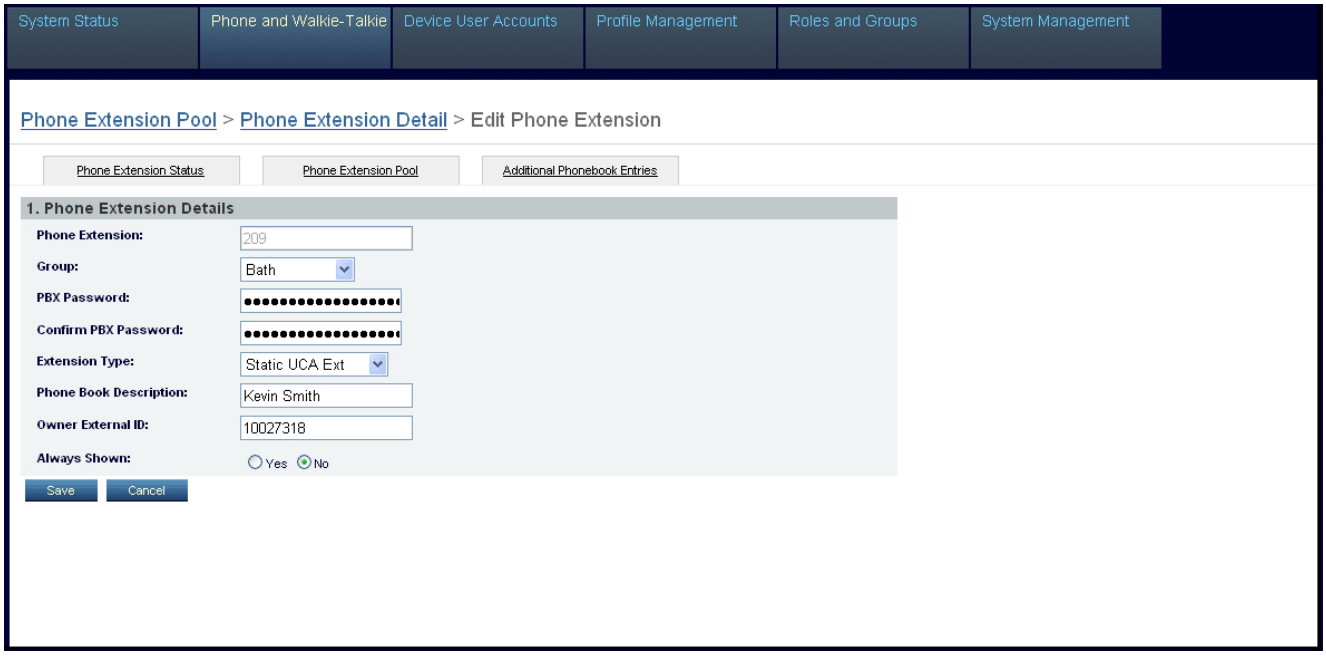


Figure 9-12 *Edit Phone Extension Window*

5. Click **Edit**.
   The SCCP *Edit Phone Extension* window displays.

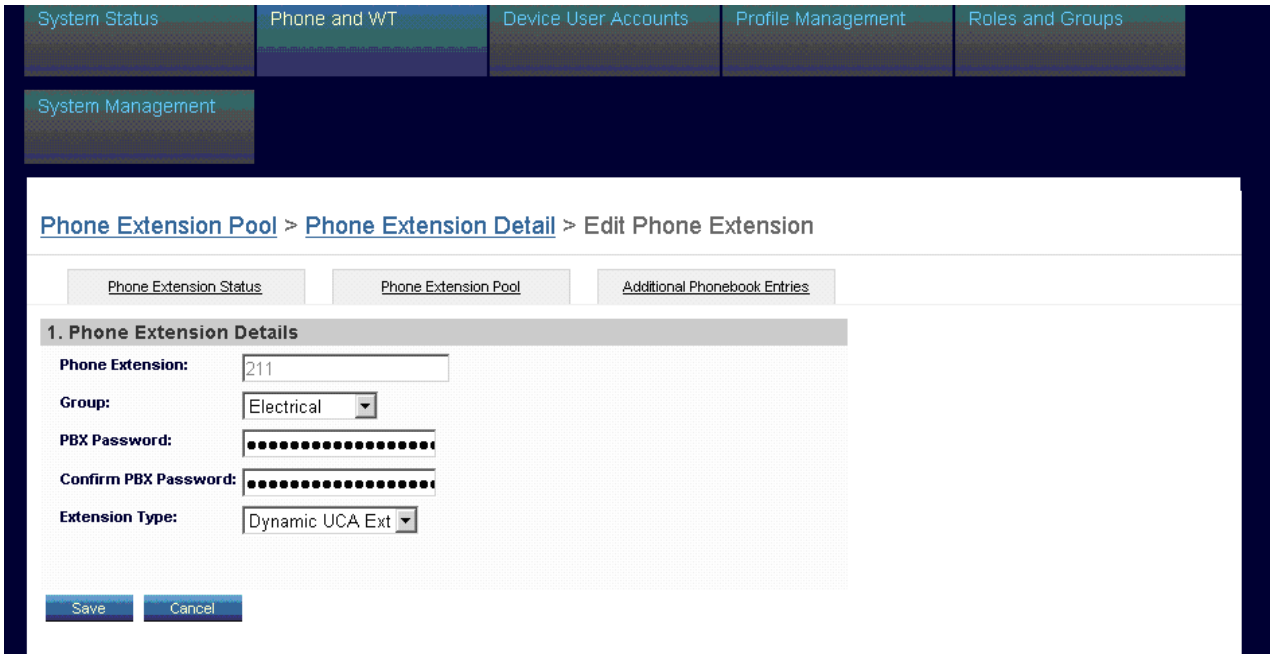


Figure 9-13 *SCCP Edit Phone Extension Window*

6. Any field, except *Phone Extension*, can be updated in this window. Enter required changes.
7. Click **Save** to save the edits or **Cancel** to terminate the session.

Phone Extension Import File Format

Use this option to load data in bulk instead of creating users and phone extensions one at a time via the UCA server's Web Console. Bulk loading of phone extensions and CA50 users into the UCA database is enabled.

To create and import user data, see *page 9-25*.

The phone extension import feature uses a comma separated value (CSV) format that must include a header line as follows:

locExtID,phnExtPhoneNumber,phnRangeToken,petID,phnExtensionPassword,phnDesc,phnIsAlwaysShown,phnOwnerID

Sample Import Lines

Sample import lines are as follows:

0002,7000,Lumber,N,Lumber Desk,Y,

Sample import line notes:

- If the location ID string has leading zeros in the database, leading zeros must be included in the import file. If the file is edited using Microsoft Excel, the field must be manually formatted to text within Excel.
- Between items, a space is optional. (The sample above shows items with and without spaces.)
- The Range Token must match the configured group tokens.
- If it is desired to use a comma in the description, it must be enclosed in quotes.
- The petID must include an N, D or S:
  - N = Additional Phonebook Entries (non-UCA)
  - D = Dynamically assigned UCA extensions (in the Phone Extension Pool)
  - S = Statically assigned UCA extensions (in the Phone Extension Pool).
- The single import option imports extensions for the Phone Extension Pool and the Additional Phonebook Entries.

NOTE For information on how to import CA50 users into the UCA database, see *page 9-25*.

Additional Phonebook Entries

The *Additional Phonebook Entries* view displays the *Phone Extension, Group* and *Description* of all additional phone extensions created in the UCA server. These phone extensions are part of the PBX System used at the location.

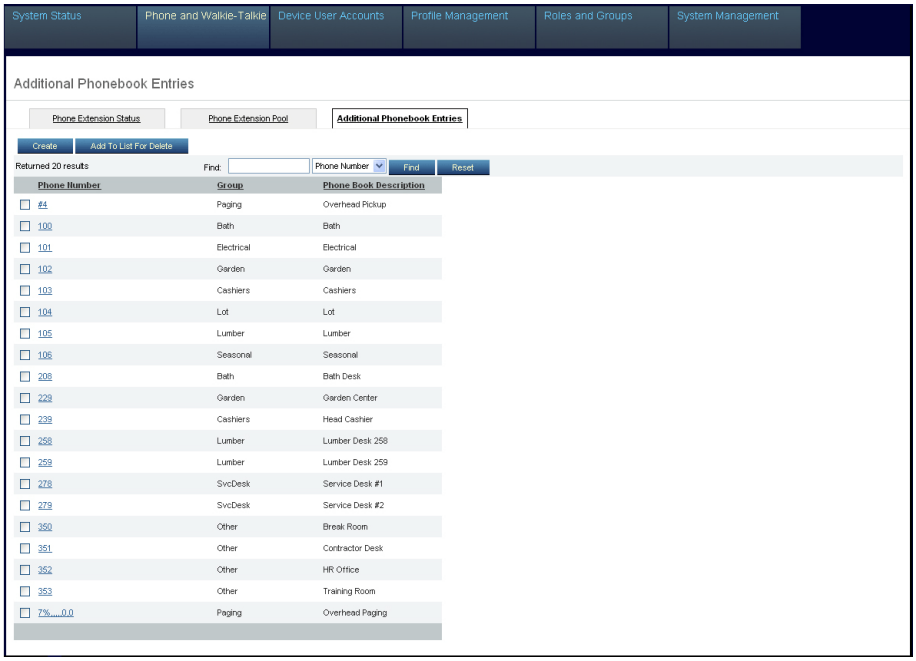


Figure 9-14 *Additional Phonebook Entries View*

Creating an Additional Phone Extension

To create an additional phone extension in the UCA server:

1. Click the *Phone and Walkie-Talkie* tab > **Additional Phonebook Entries** tab > **Create**.
   The *Additional Phonebook Entry Create* window displays.

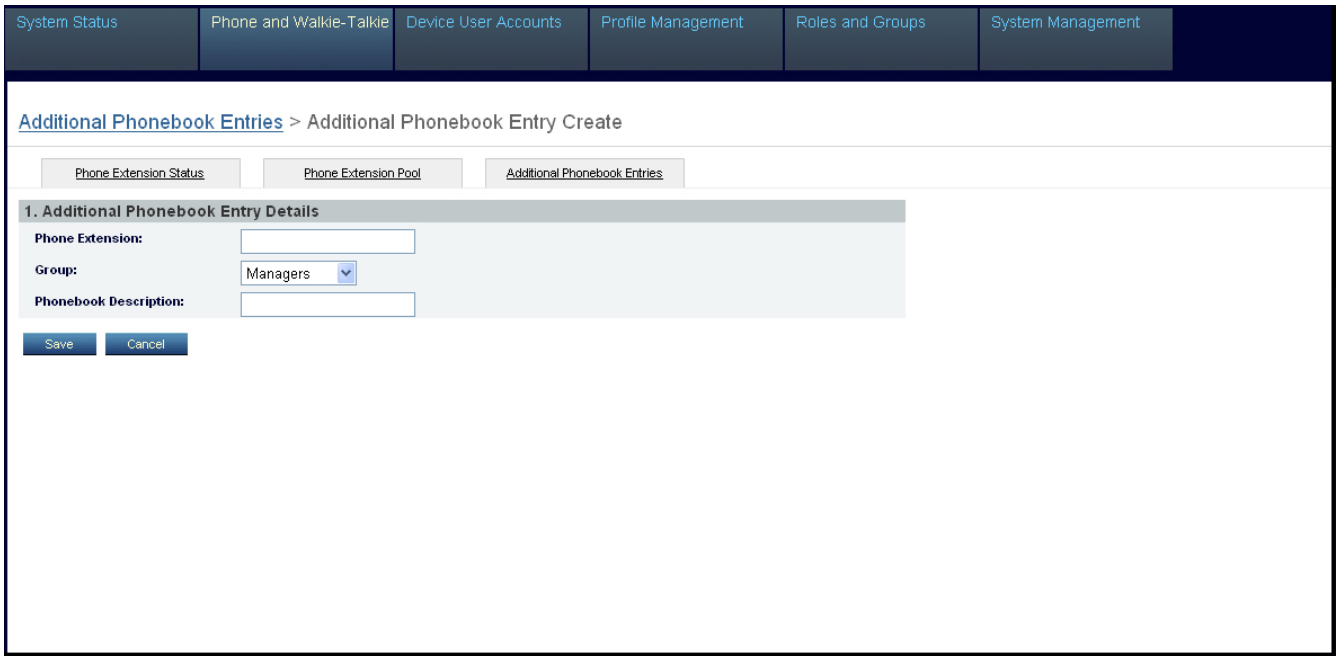


Figure 9-15 *Additional Phonebook Entry Create Window*

2. Enter a *Phone Extension* and *Phonebook Description* in the appropriate text boxes.
3. Select a *Group* from the drop down list menu.
4. Click **Save** to store the user or **Cancel** to terminate the session.

Editing an Additional Phone Extension

To edit an additional phone extension in the UCA server:

1. Click the *Phone and Walkie-Talkie* tab > **Additional Phonebook Entries** tab.
2. Click the phone extension to edit from the list of phone numbers in the *Phone Number* column (see *Figure 9-14 on page 9-14*).
   The *Additional Phonebook Entry Detail* window displays.

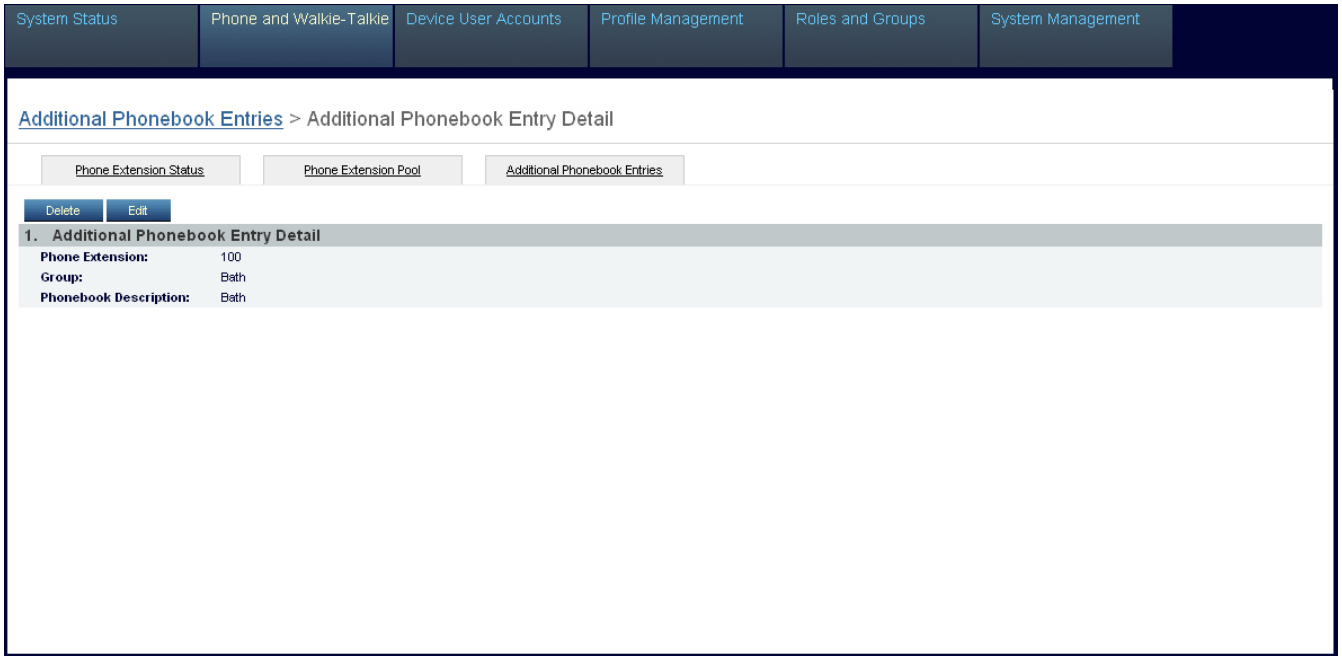


Figure 9-16 *Additional Phonebook Entry Detail Window*

3. Click **Edit**.
   The *Edit Additional Phonebook Entry* window displays.

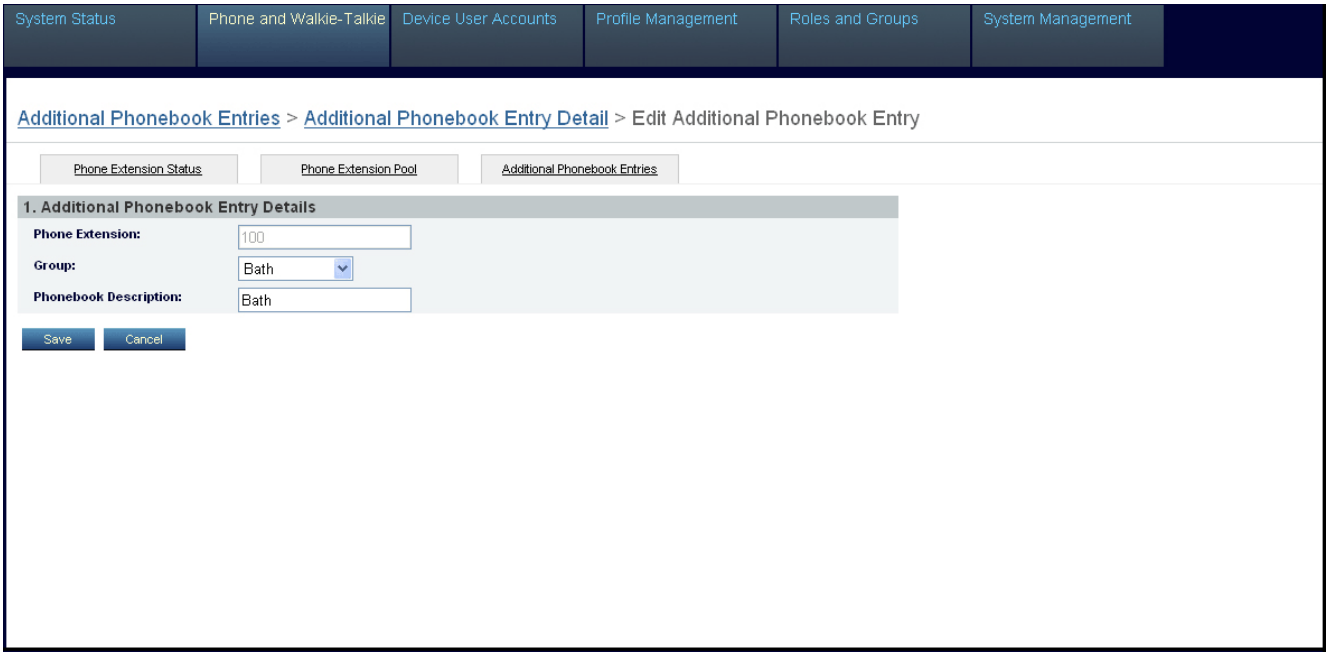


Figure 9-17 *Edit Additional Phonebook Entry Window*

4. Any field, except *Phone Extension*, can be updated in this window. Enter changes as needed.
5. Click **Save** to save the edits or **Cancel** to terminate the session.

Deleting Phone Extensions

See *Delete Utility on page 9-52* for information about deleting phone extensions.

Device User Accounts Tab View

Click the *Device User Accounts* tab on the UCA server WebConsole GUI window to display the view.

The *Device User Accounts* tab has two views:

- *Recent Users* (*Figure 9-18*)
- *All Users* (*Figure 9-19*).

Each view contains all information about users defined in the UCA server. To toggle between views, click on a tab: **Recent Users** and/or **All Users**.

Recent Users

Recent Users view displays the First/Last Name, External User ID, Role, Login Group, Assigned Phone Extension, User Login and Activity Logs of users recently seen by the UCA server. In this view create, edit and delete users.

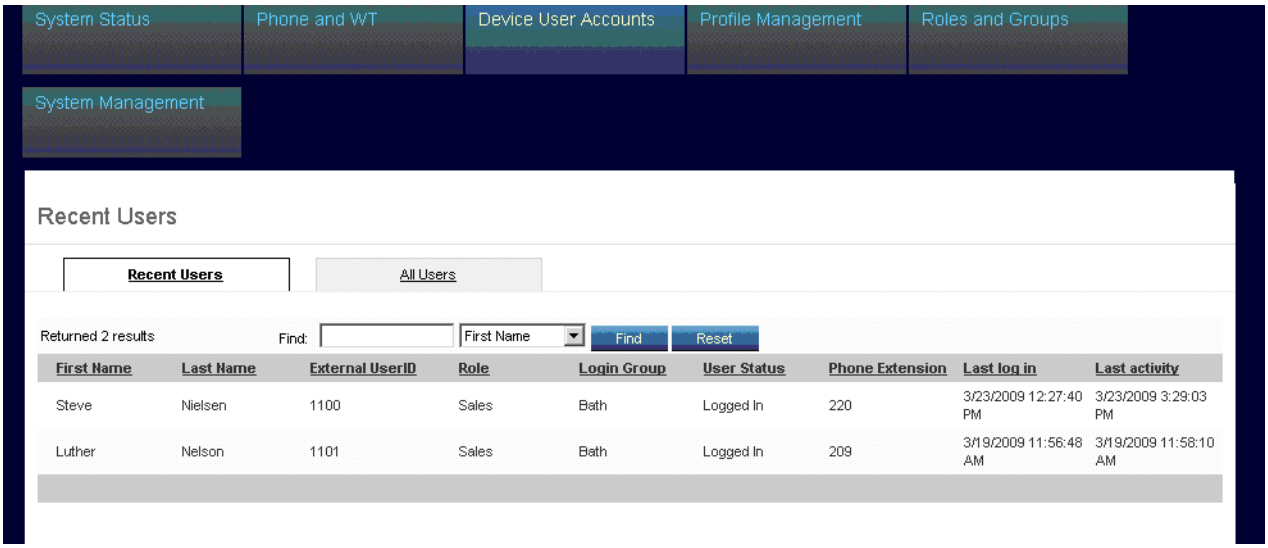


Figure 9-18 *Recent Users View*

All Users

All Users view displays the First/Last Name, Common Name, External User ID, Role, and Status of all device users configured to use the UCA server. In this view create, edit and delete users.

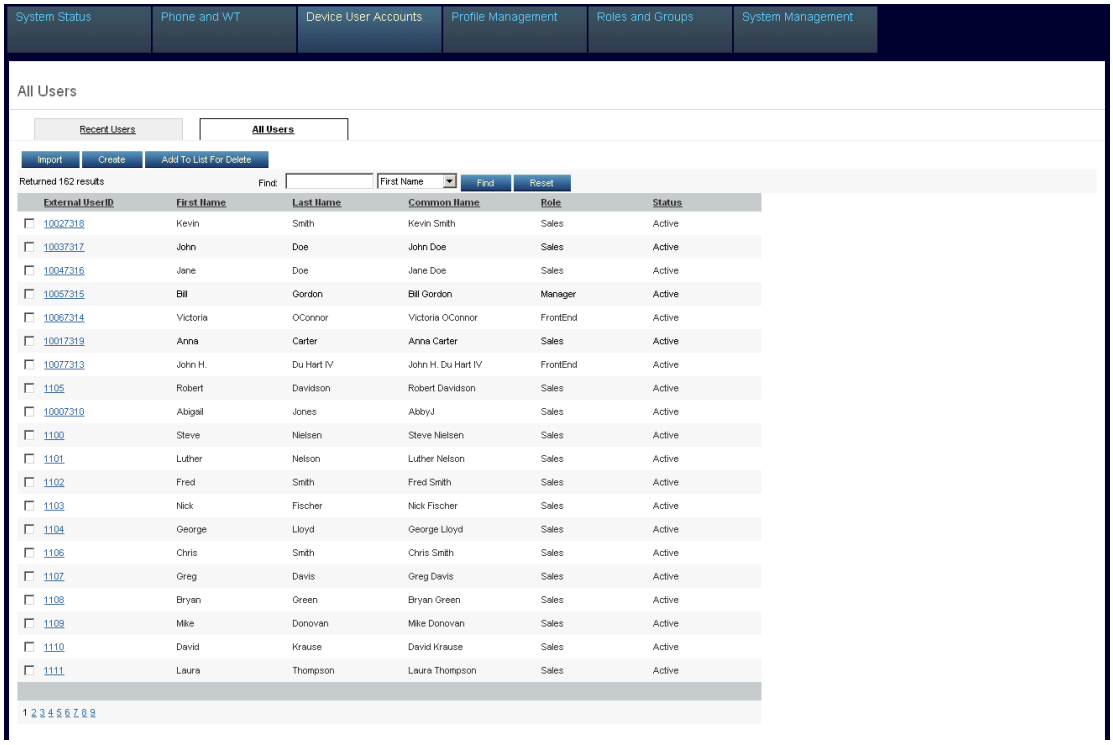

System Status
Phone and WT
Device User Accounts
Profile Management
Roles and Groups
System Management
All Users
Recent Users
All Users
Import
Create
Add To List For Delete
Returned 162 results
Find:
First Name
Find
Reset
External UserID
First Name
Last Name
Common Name
Role
Status
10027318 Kevin Smith Kevin Smith Sales Active
10037317 John Doe John Doe Sales Active
10047316 Jane Doe Jane Doe Sales Active
10057315 Bill Gordon Bill Gordon Manager Active
10067314 Victoria OConnor Victoria OConnor FrontEnd Active
10017319 Anna Carter Anna Carter Sales Active
10077313 John H. Du Hart IV John H. Du Hart IV FrontEnd Active
1105 Robert Davidson Robert Davidson Sales Active
10007310 Abigail Jones AbbyJ Sales Active
1100 Steve Nielsen Steve Nielsen Sales Active
1101 Luther Nelson Luther Nelson Sales Active
1102 Fred Smith Fred Smith Sales Active
1103 Nick Fischer Nick Fischer Sales Active
1104 George Lloyd George Lloyd Sales Active
1106 Chris Smith Chris Smith Sales Active
1107 Greg Davis Greg Davis Sales Active
1108 Bryan Green Bryan Green Sales Active
1109 Mike Donovan Mike Donovan Sales Active
1110 David Krause David Krause Sales Active
1111 Laura Thompson Laura Thompson Sales Active
1 2 3 4 5 6 7 8 9

Figure 9-19 *All Users View*

Creating a New User

To create a user in the UCA server:

1. Click the *Device User Accounts* tab > **All Users** tab > **Create**.
   The *Create User* window displays.

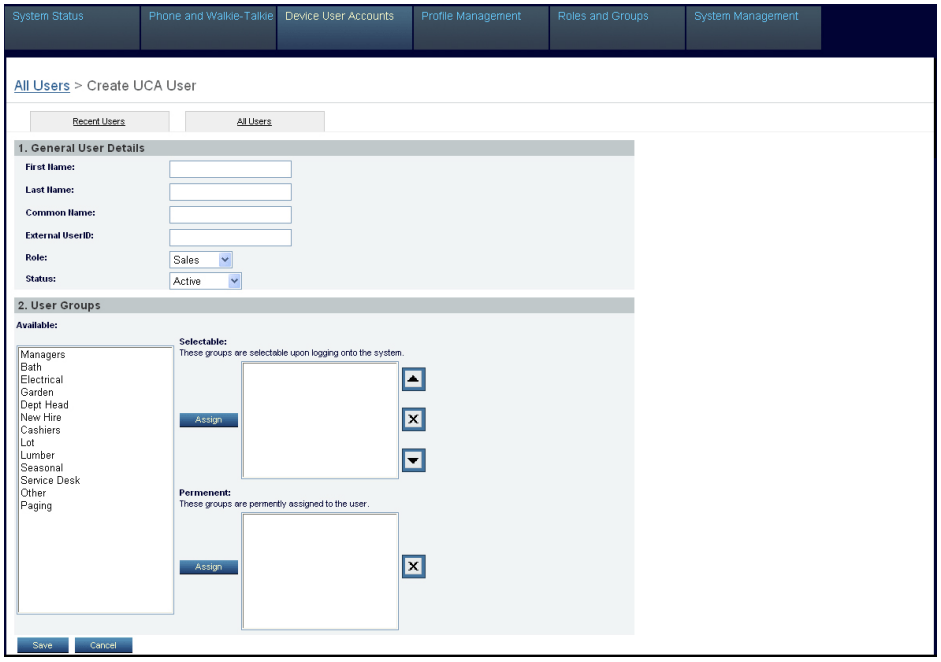


Figure 9-20 *Create User Window*

2. Enter in *First Name, Last Name, Common Name* and *External User ID* in the appropriate text boxes.

> ***NOTE*** The *External User ID* is the bar code a user scans to log in to the UCA server.

3. Select a *Role* from the drop down list menu.
4. Select a *Status* from the drop down list menu.
5. Add *Selectable* and *Permanent* assignments for the user by selecting an item in the *Available:* list and clicking **Assign** to add the item to the appropriate assignment category.

   If an incorrect assignment was added to a group, click **X** to delete.

> ***NOTE*** The set of Selectable Groups allows a user to select a primary group at login. The set of Permanent Groups allows a user to always display as part of a specific group. All users must be assigned to at least one group (selectable or permanent).

6. Click **Save** to store the user or **Cancel** to terminate the session.

Editing User Information

To edit user information in the UCA server:

1. Click the *Device User Accounts* tab > **All Users** tab.
2. Click the external user ID to edit from the list of users in the *External UserID* column (see *Figure 9-19 on page 9-19*).
   The *UCA User* detail window displays.

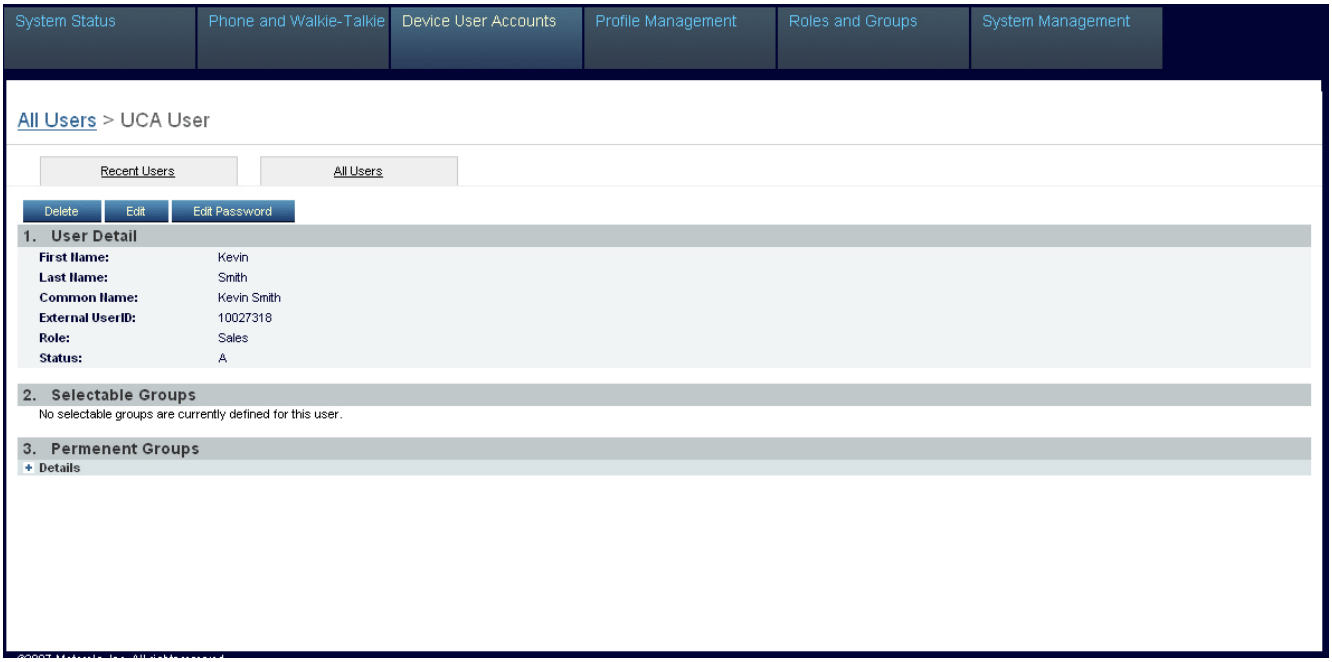


Figure 9-21 *UCA User Window*

3. Click **Edit**.
   The *Edit Phone Extension* window displays.

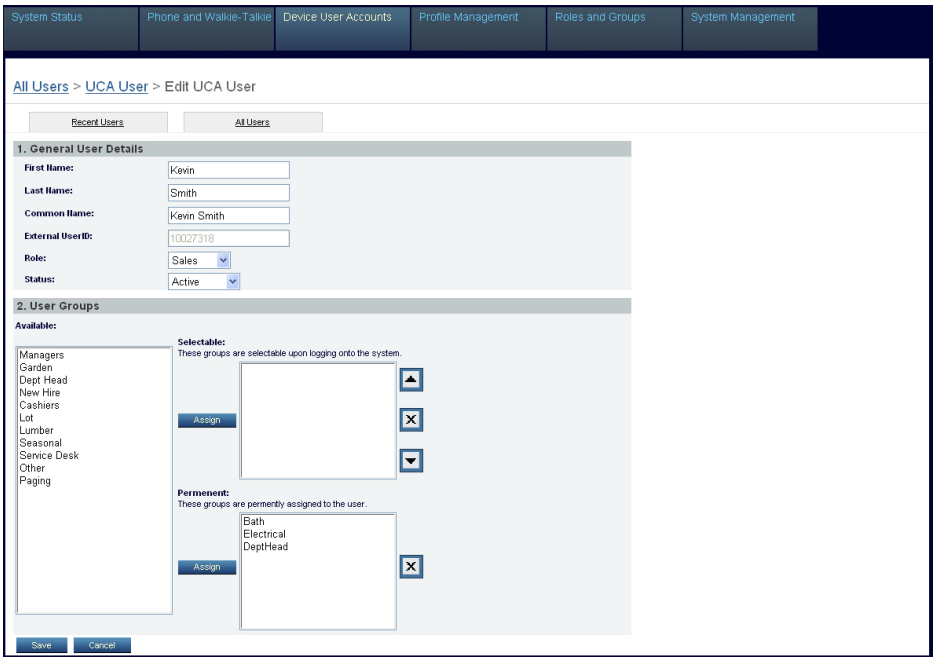


Figure 9-22 *Edit User Information Window*

4. Enter the user information to change in the appropriate text boxes; add and/or delete *Selectable* and *Permanent* assignments. as needed.

5. Click **Save** to save the edits or **Cancel** to terminate the session.

Editing User Password

A password is used during a user login to the UCA server. Password requirements are at the discretion of the UCA server Administrator. If a password is not assigned to a user, the user is not prompted to enter a password during the login process on a device.

To edit a user's password in the UCA server:

1. Click the *Device User Accounts* tab > **All Users** tab.
2. Click the external user ID to edit from the list of users in the *External UserID* column (see *Figure 9-19 on page 9-19*).
   The *UCA User* detail window displays.

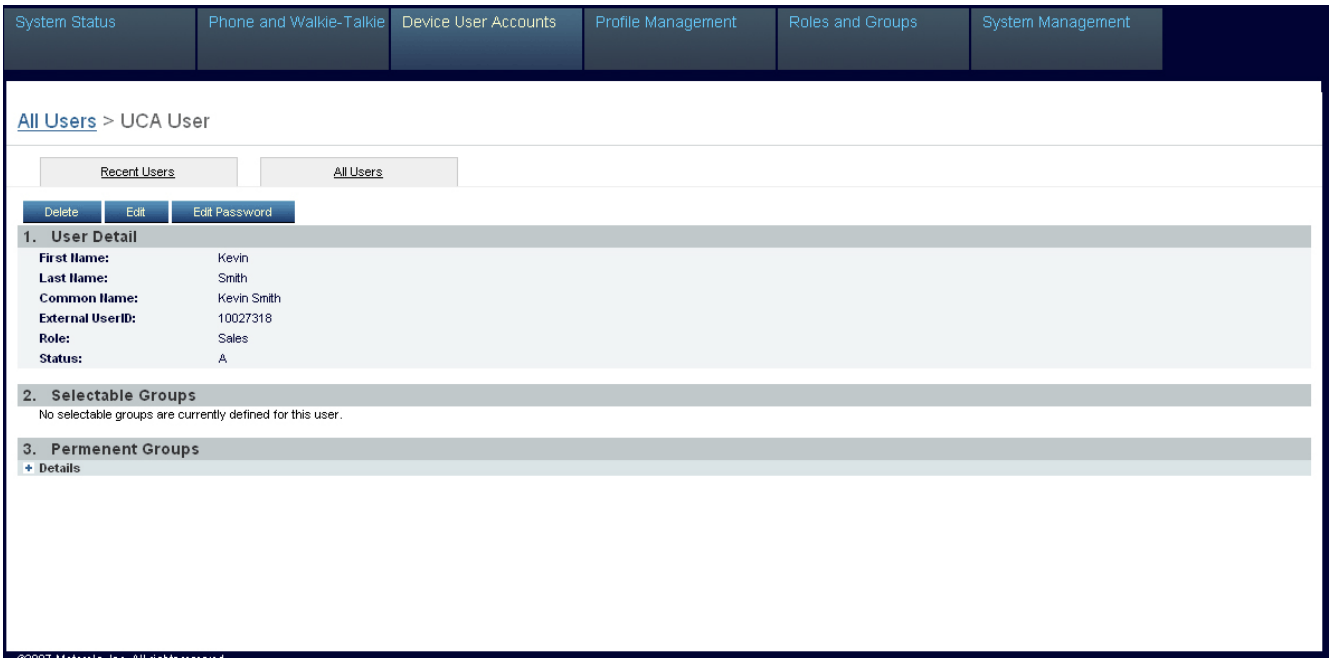


Figure 9-23 *UCA User Window*

3. Click **Edit Password**.
   The *Edit UCA User Password* window displays.

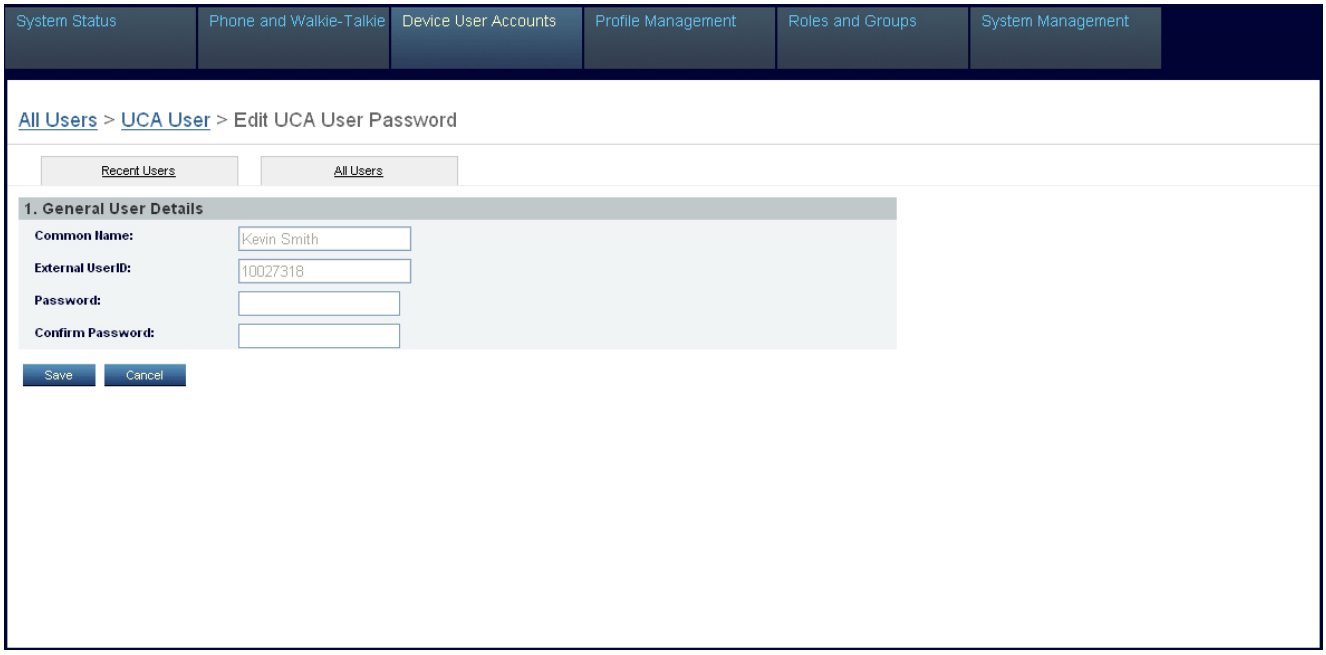


Figure 9-24 *Edit UCA User Password Window*

4. Enter the new password in the *Password:* text box.
5. Confirm the new password in the *Confirm Password:* text box.
6. Click **Save** to change the password or **Cancel** to terminate the session.

Deleting a User

See *Delete Utility on page 9-52* for information about deleting entries in the UCA server.

Device User Import File Format

Use this option to load data in bulk instead of creating users and phone extensions one at a time via the UCA server's Web Console. Bulk loading of phone extensions and CA50 users into the UCA database is enabled.

To create and import phone extensions, see *page 9-13*.

The CA50 user import feature uses a comma separated value (CSV) format that must include a header line as follows:

locExtID,ExtUsrID,firstName,lastName,userName,PIN,RoleToken,userStatus,Selectable Group List,Permanent Group List

Sample Import Line

A sample import line is as follows:

0002,2000, Joe,Smith,Joe Smith, 5555,Sales,A,"Lumber,Electrical,Bath",NewHire

Sample import line notes:

- If the location ID string has leading zeros in the database, leading zeros must be included in the import file. If the file is edited using Microsoft Excel, the field must be manually formatted to text within Excel.
- Between items, a space is optional. (The sample above shows items with and without spaces.)
- The Selectable and Permanent Group Lists must be enclosed in quotes since the groups in the lists are comma separated.
- The group token is used in the group lists.
- The user status is typically 'A' (active). Other options are 'D' (disabled) and 'S' (suspended).
- It is recommended to start with the sample import file found in the Development Toolkit folder.

NOTE For information on how to import phone extensions into the UCA database, see *page 9-13*.

Profile Management Tab View

Click the *Profile Management* tab on the UCA server WebConsole GUI window to display the view.

Profile Management provides the ability to associate a base XML profile template to a *Role* and/or *Group* entity. The base XML profile is a file that contains configuration settings to set up a UCA user when they first login to the UCA server. When a user logs into the server, the login value is compared to a *Role* and/or *Group* entity. The *Role* and/or *Group* entity is then used to associate a base XML profile template for that user. The UCA server uses this base profile as a starting point to build a complete unique user profile (XML file) that is sent back to the UCA client application.

To view associated base profiles to a *Role* entity in the UCA server, click the *Profile Management* tab > **Profile Associations**.

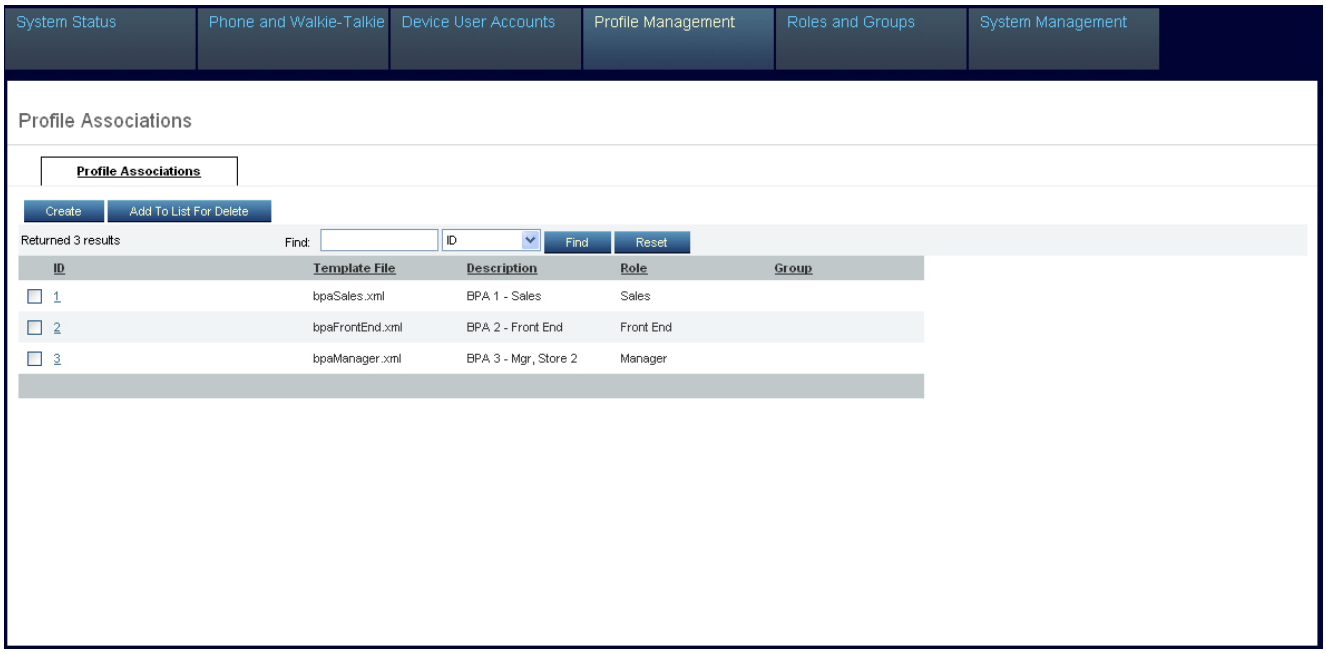


Figure 9-25 *Profile Management View*

Create an Associated Base Profile

To create an associated base profile to a *Role* entry in the UCA server:

1. Click the *Profile Management* tab > **Profile Associations** tab > **Create**.
   The *Profile Association Create* window displays.

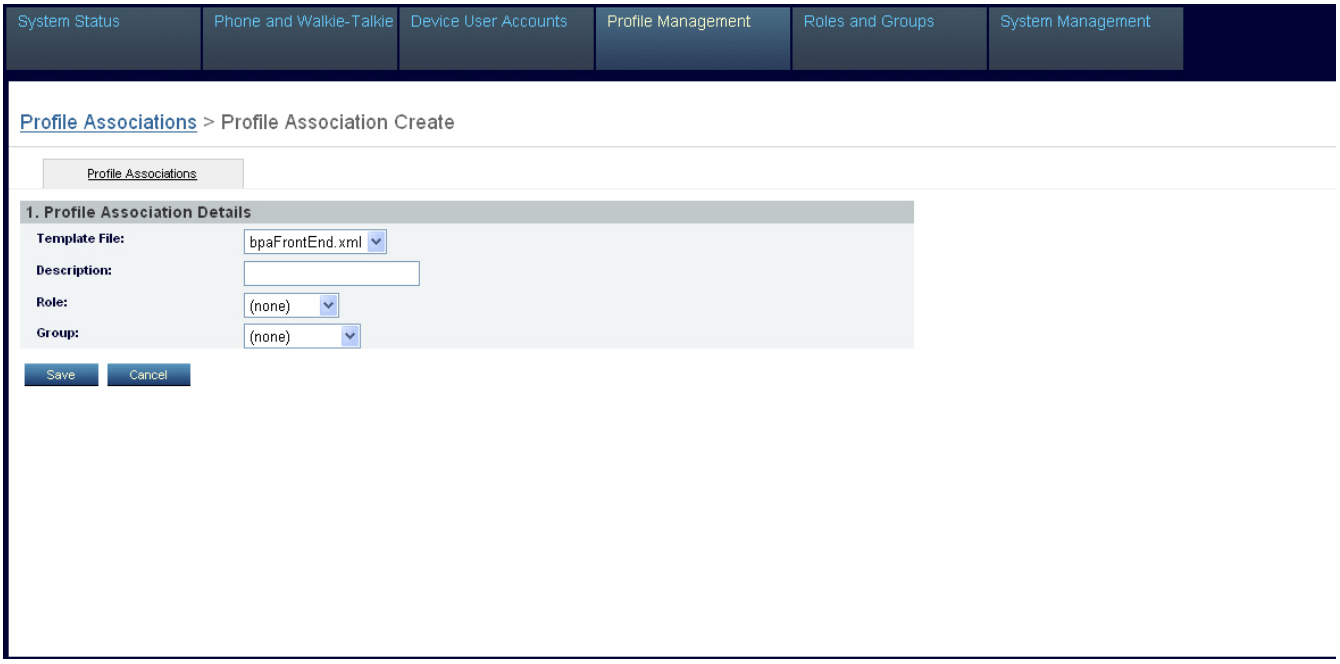


Figure 9-26 *Profile Association Create Window*

2. Select a template from the *Template File:* drop down list menu. The template is an actual base profile (XML file).

> ***NOTE*** Templates are found in the directory specified by the system setting: *System.Profile.Template.Path*.

3. Enter a description in the *Description:* text box.
4. Select a role to associate with this profile from the *Role:* drop down list menu.
5. Select a group to associate with this profile from the *Group:* drop down list menu.
6. Click **Save** to store the group or **Cancel** to terminate the session.

Editing an Associated Base Profile

To edit an associated base profile to a *Role* entry in the UCA server:

1. Click the *Profile Management* tab.
2. Select an ID (base profile association number) in the list under the *ID* column.
   The *Profile Association Detail* window displays.

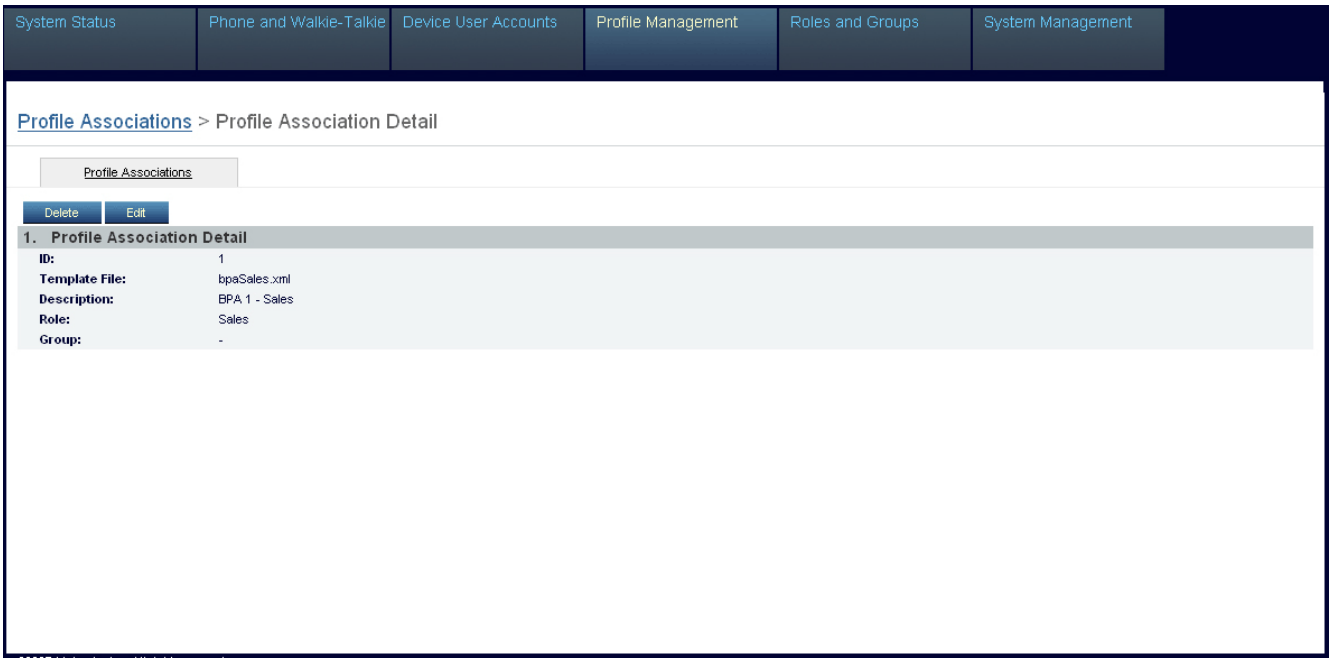


Figure 9-27 *Profile Association Detail Window*

3. Click **Edit**.
   The *Edit Profile Association* window displays.

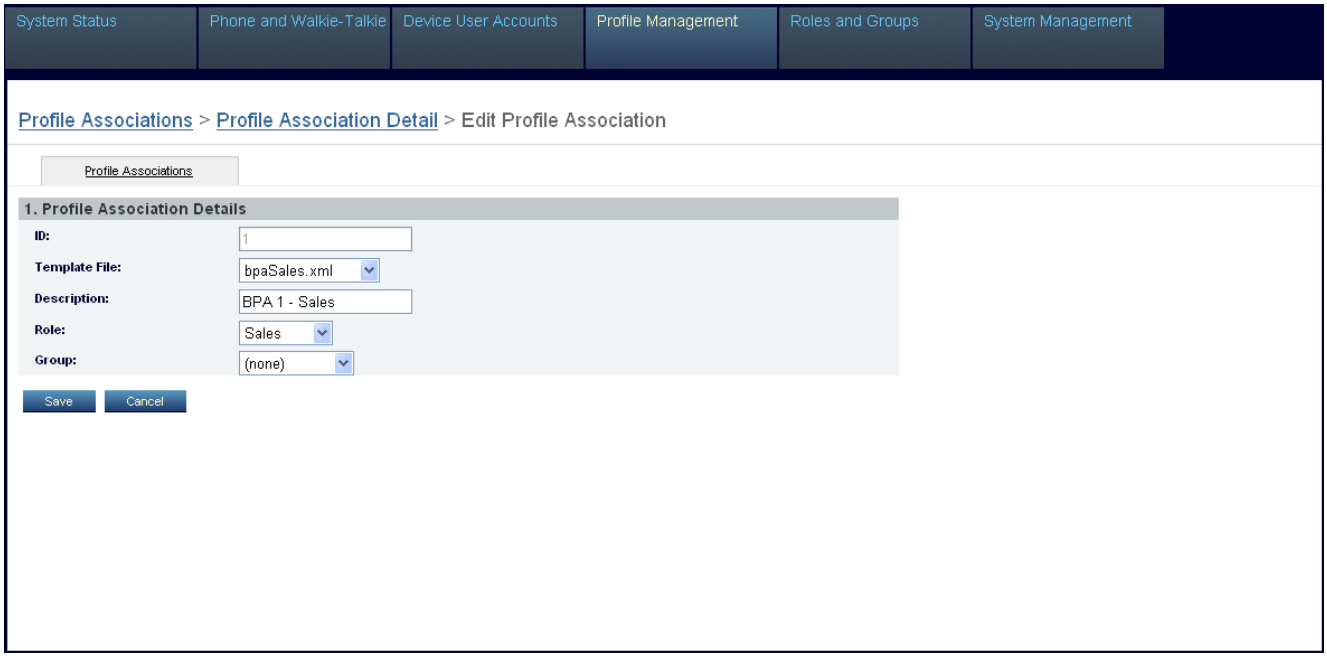


Figure 9-28 *Edit Associated Base Profile Window*

4. Enter a new description in the *Description:* text box; select a role and group from the drop down list menus as needed.
5. Click **Save** to change the associated base profile or **Cancel** to terminate the session.

Deleting Associated Base Profiles

See *Delete Utility on page 9-52* for information about deleting entries in the UCA server.

Roles and Groups Tab View

Click the *Roles and Groups* tab on the UCA server WebConsole GUI window to display the view.

The *Roles and Groups* tab has two views:

- *Groups* (*Figure 9-29*)
- *Roles* (*Figure 9-33*).

Each view contains all information about groups and roles in the server. To toggle between views, click on a tab: **Groups** and/or **Roles**.

Groups

A *Group* contains any number of users comprising a unit, such as a department. For example, the Electrical Department in a store would include several employees who can be users comprising a *Group*.

Groups view displays the current groups in the server. In this view create, edit and delete groups.

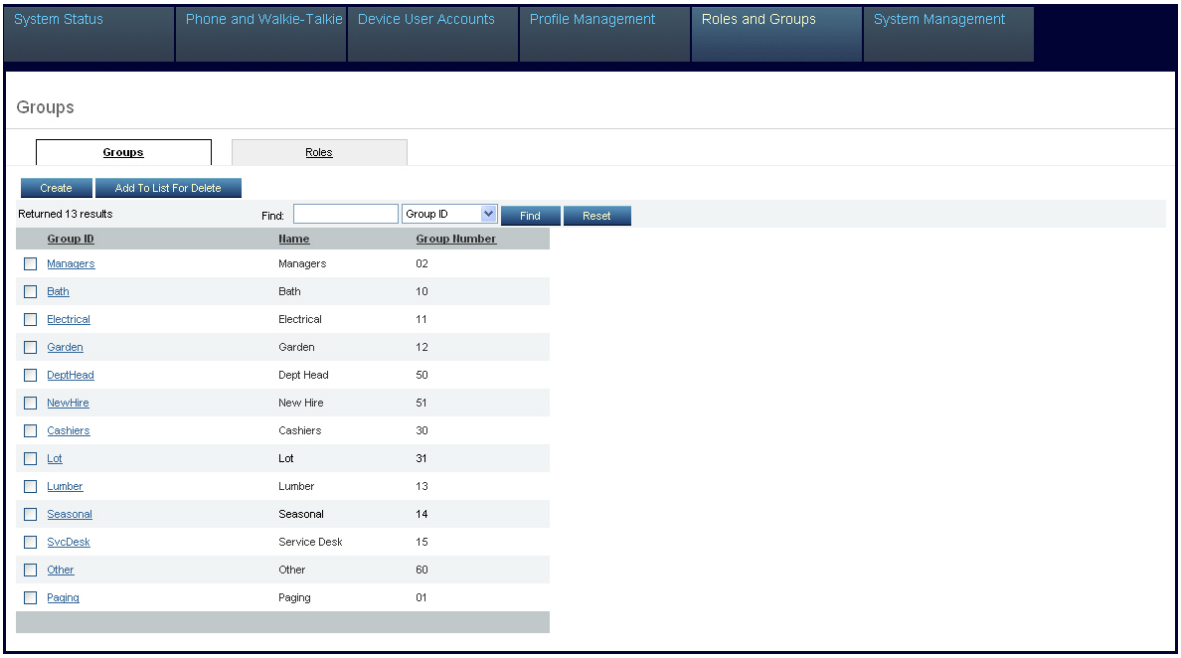


Figure 9-29 *Groups View*

Creating Groups

To create a group in the UCA server:

1. Click the *Roles and Groups* tab > **Groups** tab > **Create**.
   The *Group Create* window displays.

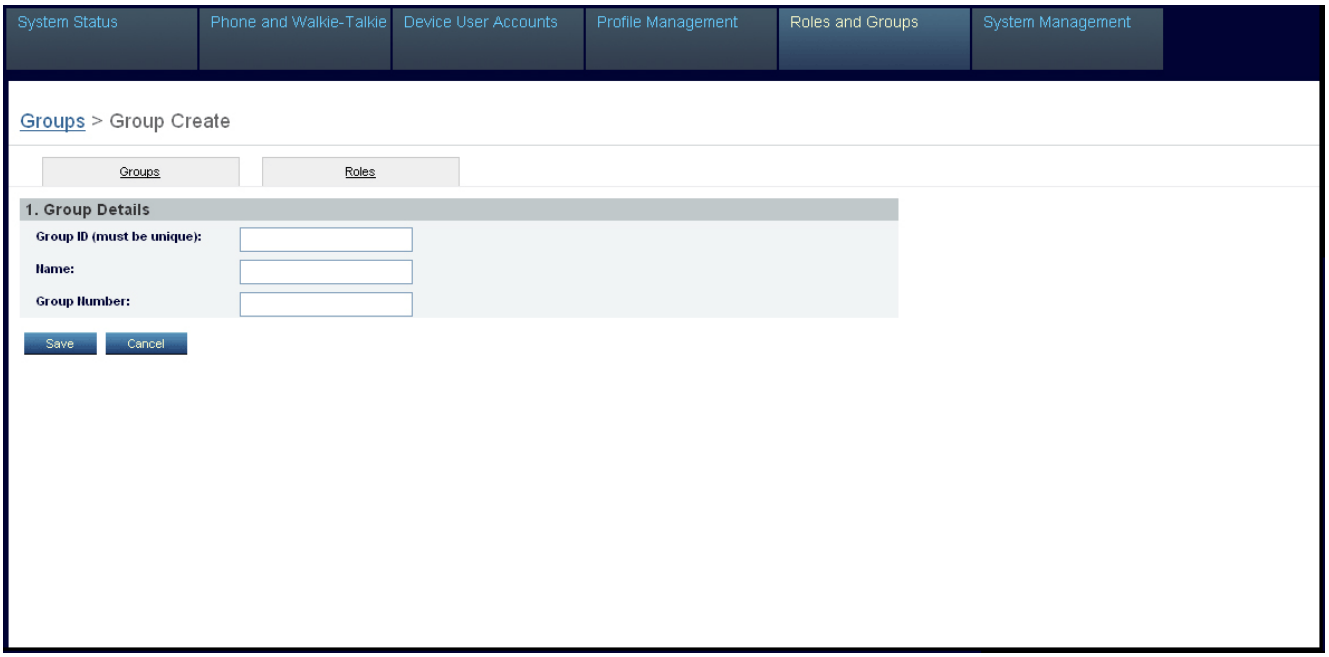


Figure 9-30 *Group Create Window*

2. Enter a unique Group ID in the *Group ID (must be unique):* text box. A *Group ID* can be a department name or abbreviation for a like group.

   NOTE Ensure you use a unique string of characters that does not include spaces. The *Group ID* is the key used internally to refer to this specific group.

3. Enter a name in the *Name:* text box. *Name* is a user friendly group name that displays to the user if the user is given a group choice at login.

4. Enter group number information in the *Group Number:* text box. The *Group Number* is used in certain situations to sort the list. This identifier might be a department number used within a company.

5. Click **Save** to store the group or **Cancel** to terminate the session.

Editing a Group

To edit a group in the UCA server:

1. Click the *Roles and Groups* tab > **Groups** tab.
2. Click the group to edit from the list of UCA groups in the *Group ID* column (see *Figure 9-29 on page 9-30*).
   The *Group Detail* window displays.

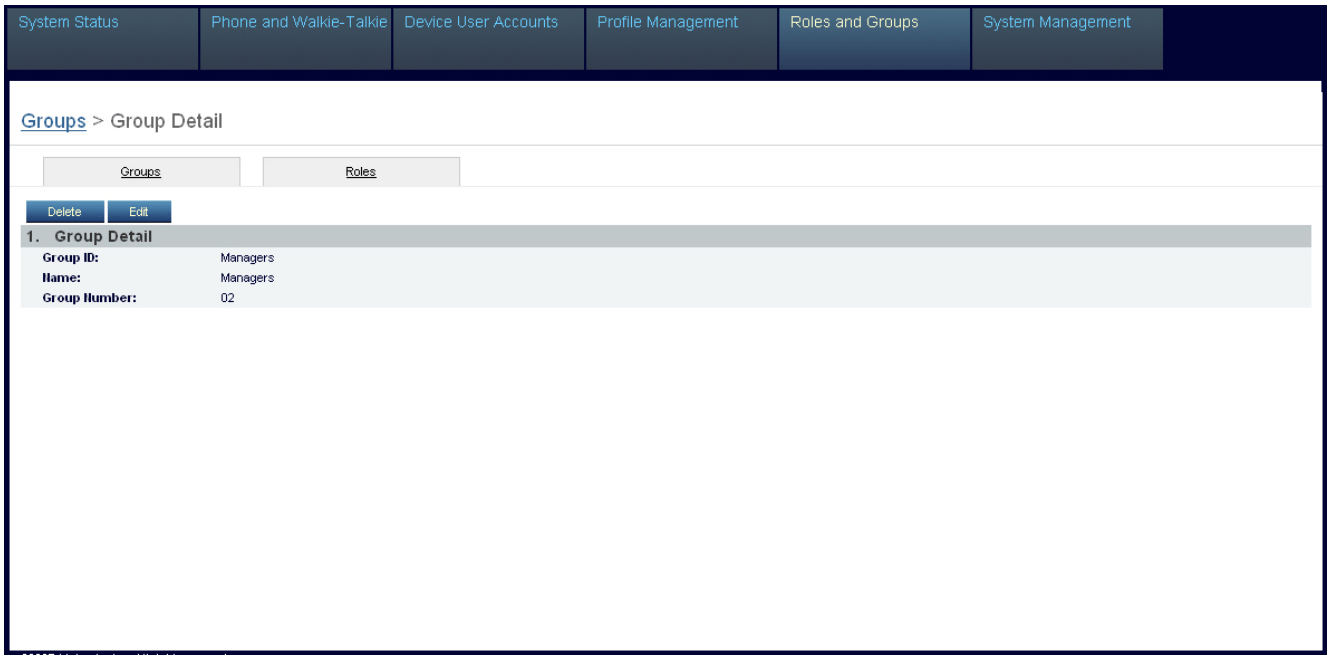


Figure 9-31 *Group Detail Window*

3. Click **Edit**.
   The *Edit Group* window displays.

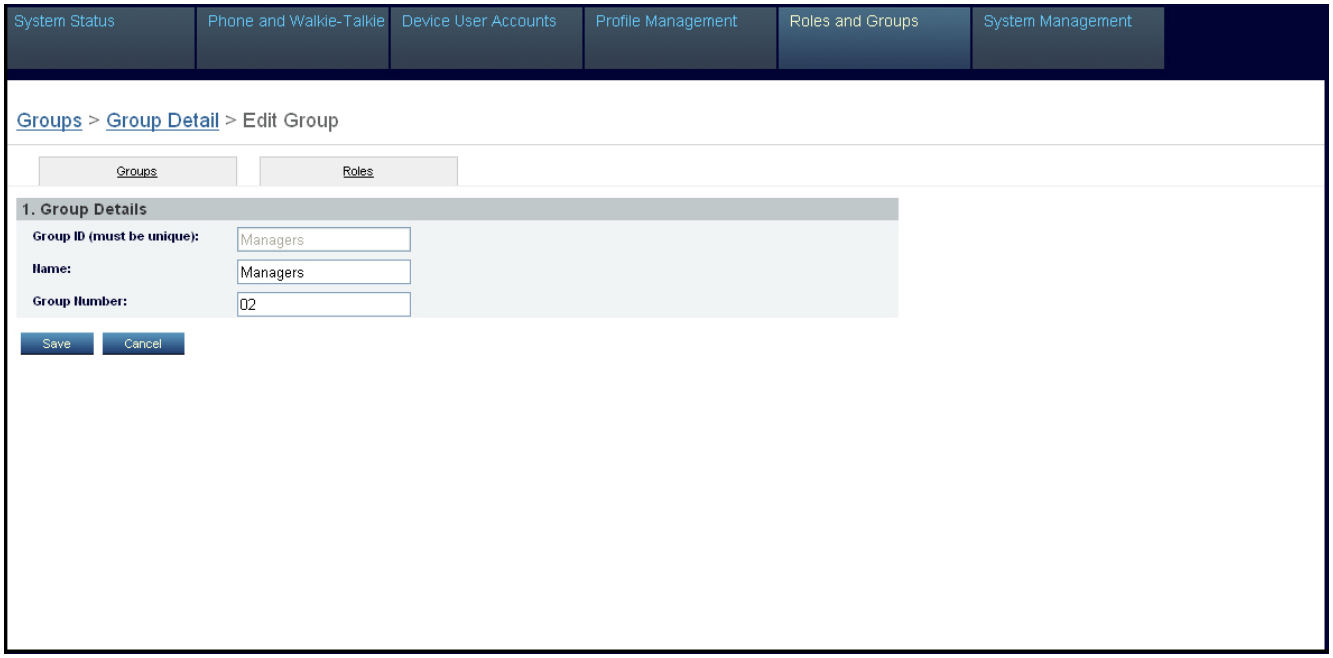


Figure 9-32 *Edit Group Window*

4. Enter the group information to change in the appropriate text boxes. Any field, except *Group ID*, can be updated in this window.
5. Click **Save** to save the edits or **Cancel** to terminate the session.

Deleting Groups

See *Delete Utility on page 9-52* for information about deleting entries in the UCA server.

Roles

Roles describe the actions and activities of each user. For example, a role can be a title such as Sales Manager or Sales Representative. *Roles* view displays the current roles in the server. In this view create, edit and delete roles.

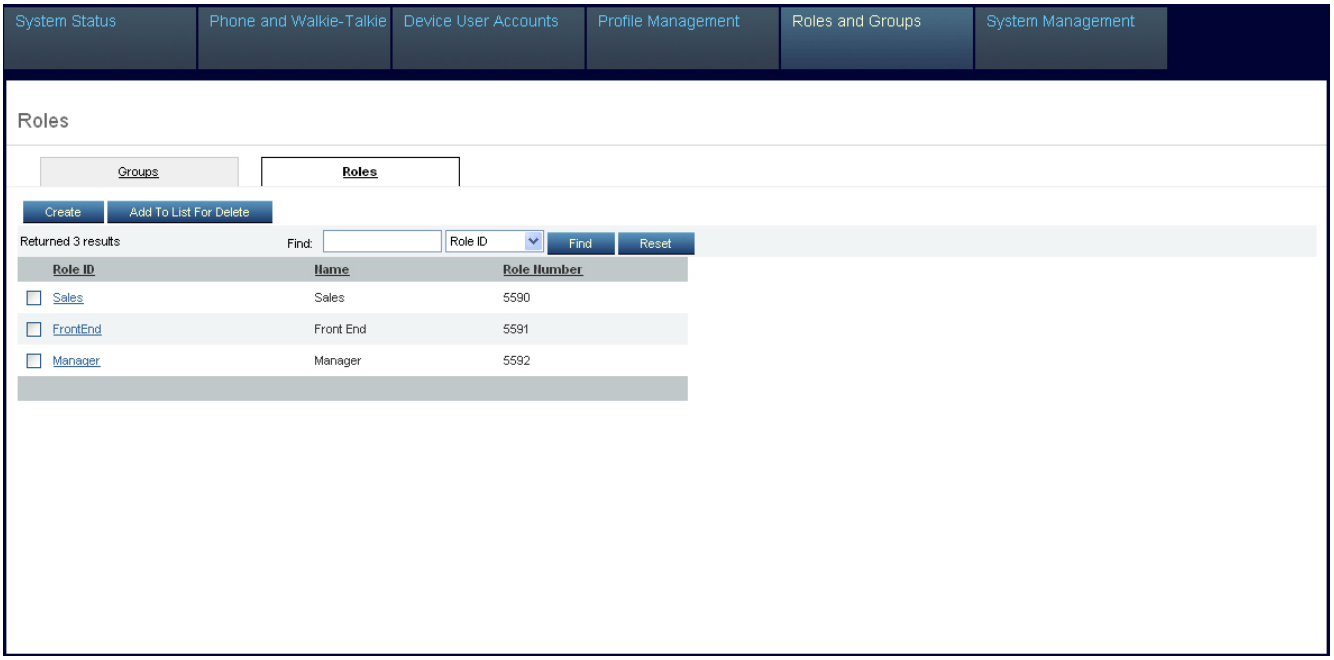


Figure 9-33 *Roles View*

Creating Roles

To create a role in the UCA server:

1. Click the *Roles and Groups* tab > **Roles** tab > **Create**.
   The *Role Create* window displays.

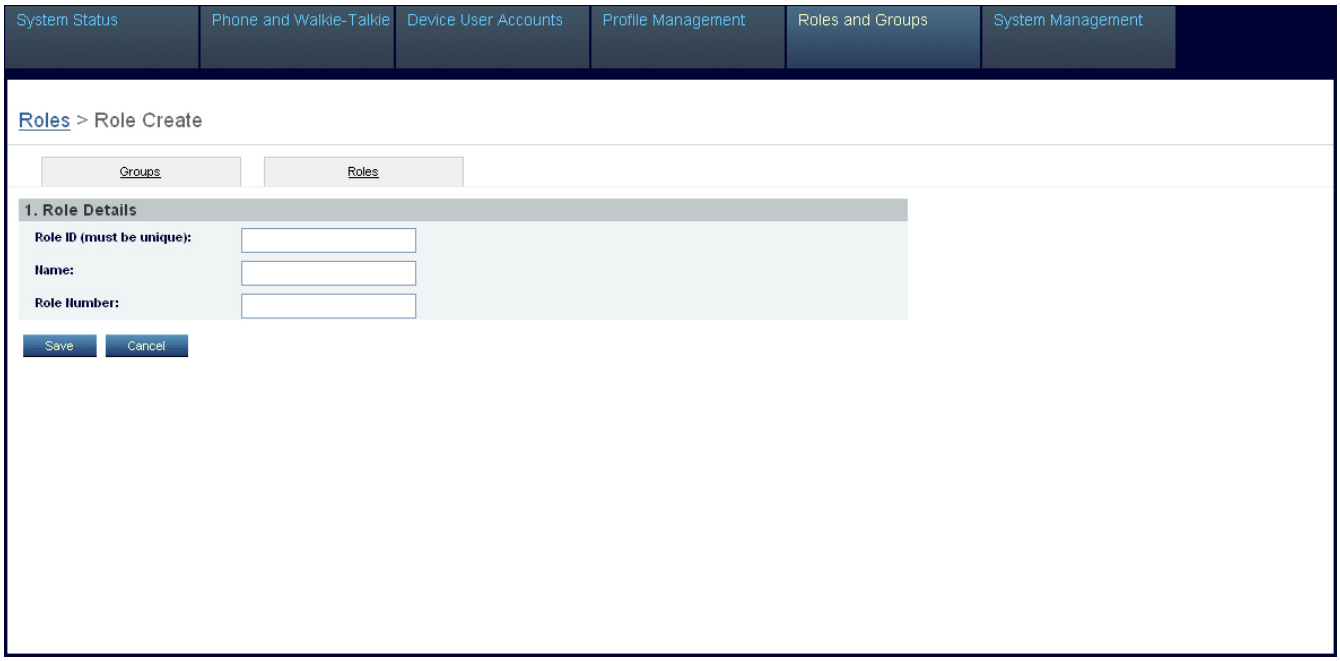


Figure 9-34 *Role Create Window*

2. Enter a unique Role ID in the *Role ID (must be unique):* text box. A *Role ID* can be an associate title.

> ***NOTE*** Ensure you use a unique string of characters that does not include spaces. The *Role ID* is the key used internally to refer to this specific role.

3. Enter a name in the *Name:* text box. *Name* is a user friendly role name that displays to the user, if necessary.
4. Enter role number information in the *Role Number:* text box. The *Role Number* is used in certain situations to sort the list.
5. Click **Save** to store the group or **Cancel** to terminate the session.

Editing a Role

To edit a role in the UCA server:

1. Click the *Roles and Groups* tab > **Role** tab.
2. Click the role to edit from the list of UCA roles in the *Role ID* column (see *Figure 9-33 on page 9-33*).
   The *Role Detail* window displays.

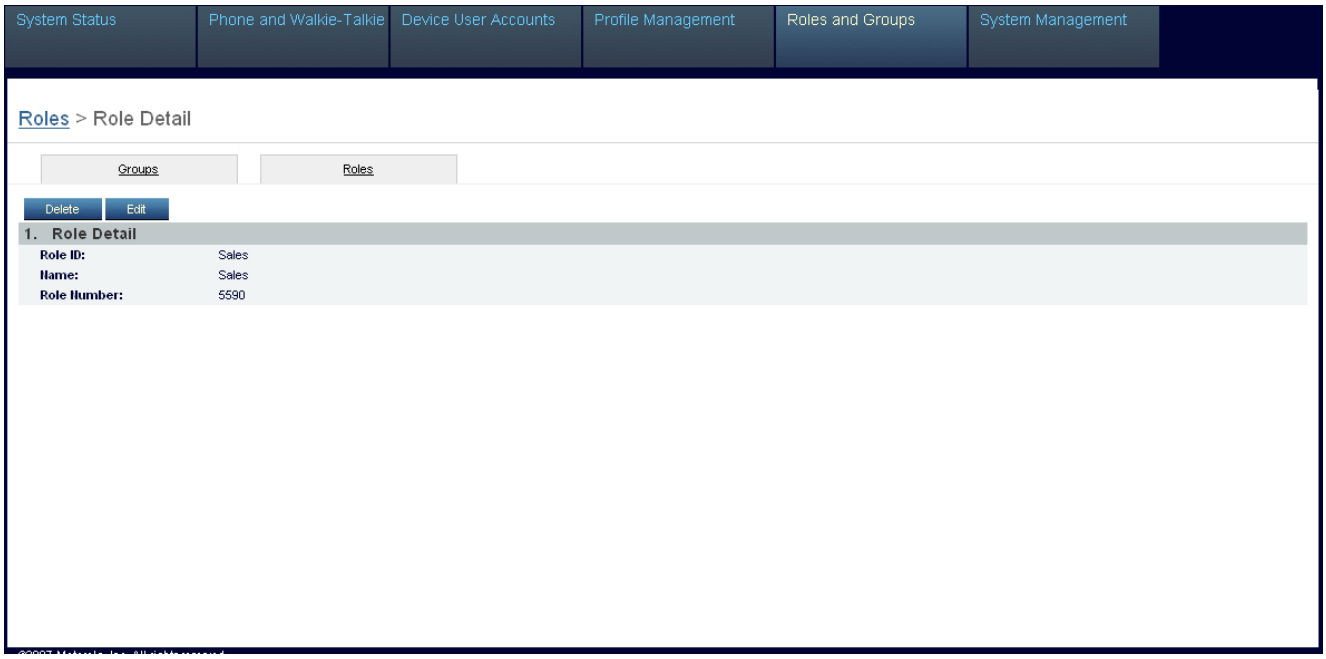


Figure 9-35 *Role Detail Window*

3. Click **Edit**.
   The *Edit Role* window displays.

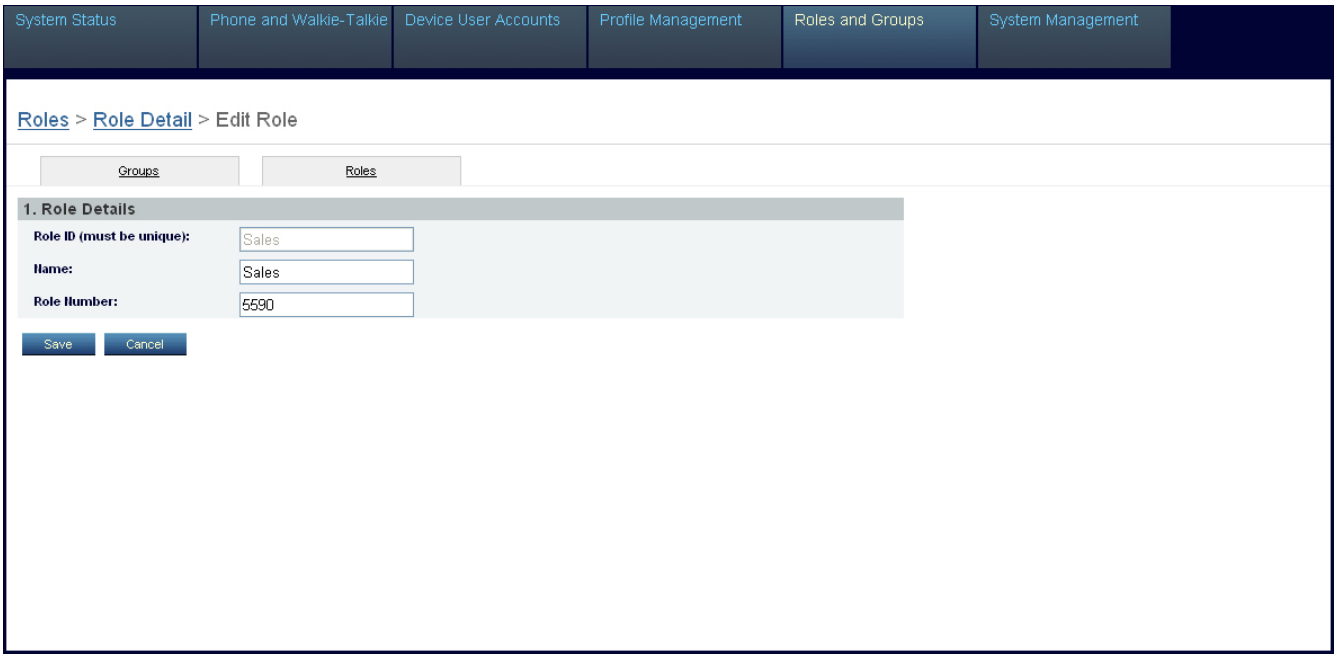


Figure 9-36 *Edit Role Window*

4. Enter the role information to change in the appropriate text boxes. Any field, except *Role ID*, can be updated in this window.
5. Click **Save** to save the edits or **Cancel** to terminate the session.

Deleting a Role

See *Delete Utility on page 9-52* for information about deleting entries in the UCA server.

System Management Tab View

Click the *System Management* tab on the UCA server WebConsole GUI window to display the view.

The *System Management* tab has three views:

- *System Settings* (*Figure 9-37*)
- *Console Users* (*Figure*)
- *Tools* (*Figure*).

Each view provides the ability to view and edit UCA System server settings.

System Settings

System Settings provides the ability to view and edit UCA System server settings. System Settings are divided into four categories:

- Client (see *Client Settings on page 9-40*)
- Phone (see *Phone Settings on page 9-41*)
- Phonebook (see *Phonebook Settings on page 9-41*)
- System (see *High Level System Settings on page 9-42*).

NOTE For a list of all system management settings, see *page 6-48*.

CAUTION Only fully trained and certified UCA Administrators should view and/or configure values within System Settings.

To view/configure System Settings:

- Click the *System Settings* tab on the UCA server WebConsole GUI window to display the tab view.

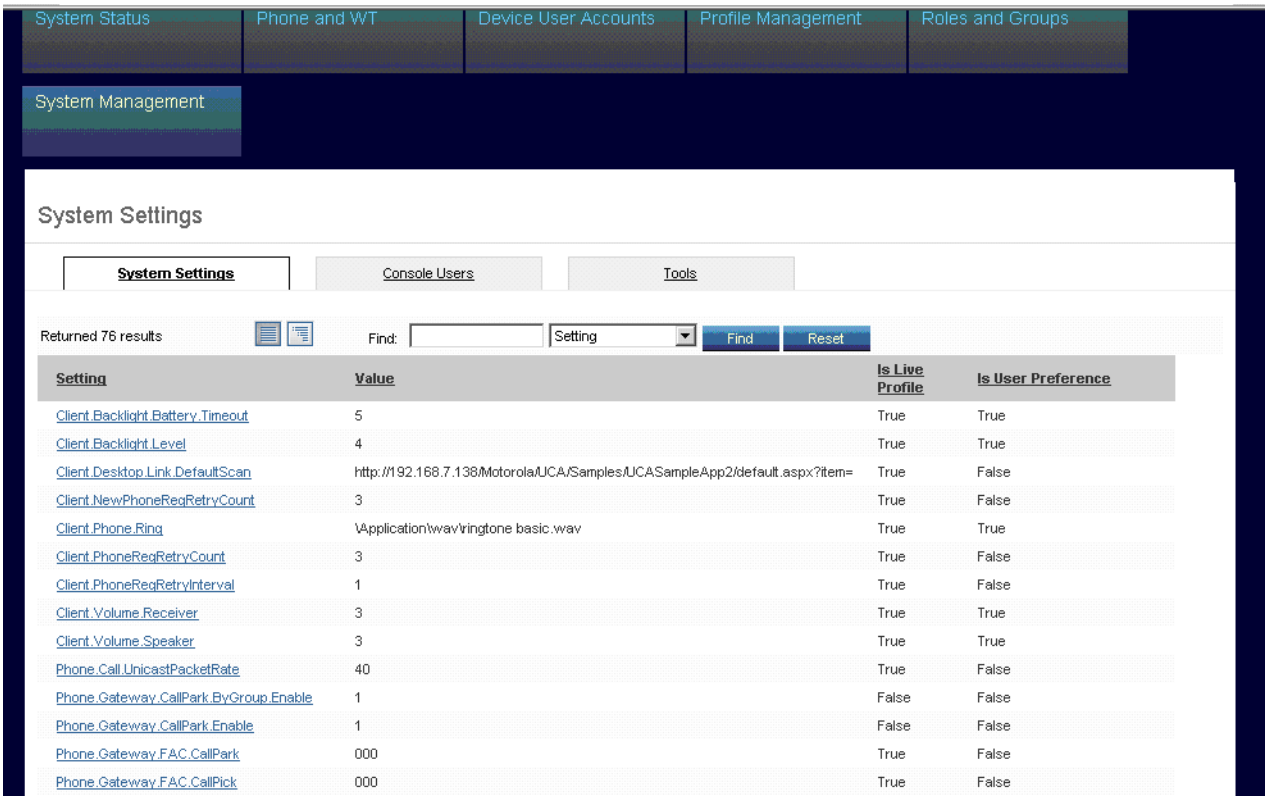


Figure 9-37 *System Settings View*

- Click to view *System Settings* in a tree structure.

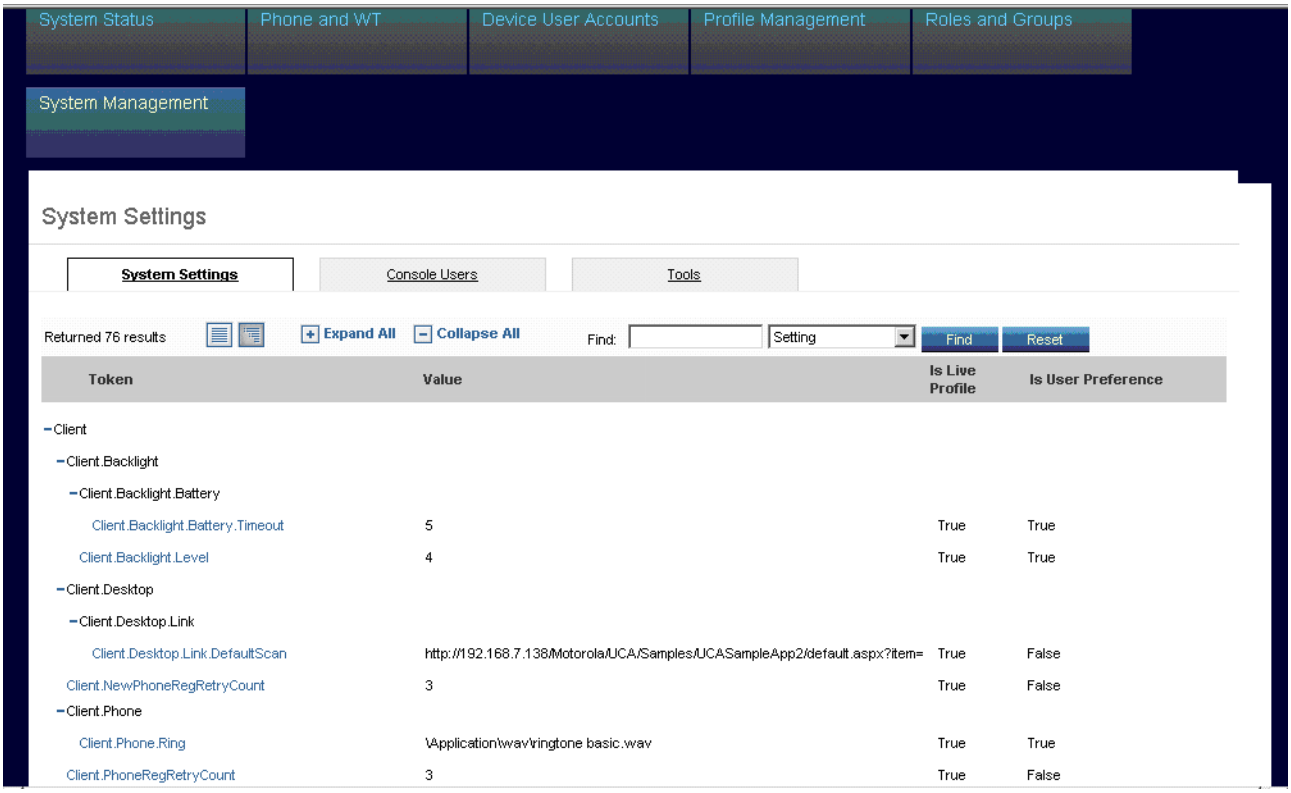


Figure 9-38 *System Settings - Tree Structure View*

NOTE See *Configure System Settings on page 9-42* for details about configuring system settings.

Client Settings

Client settings are default CA50 device settings used to help build the XML profile for a user. The user can modify certain settings on the device such as backlight, phone ring and volume settings as personal preferences. These preferences are recorded and applied when the user logs into the UCA server.

NOTE The client settings in *System Settings* represent the default values for users before they are changed during device operation.

For additional information see *page 7-1* in *Chapter 7, XML Profile for Configuration*.

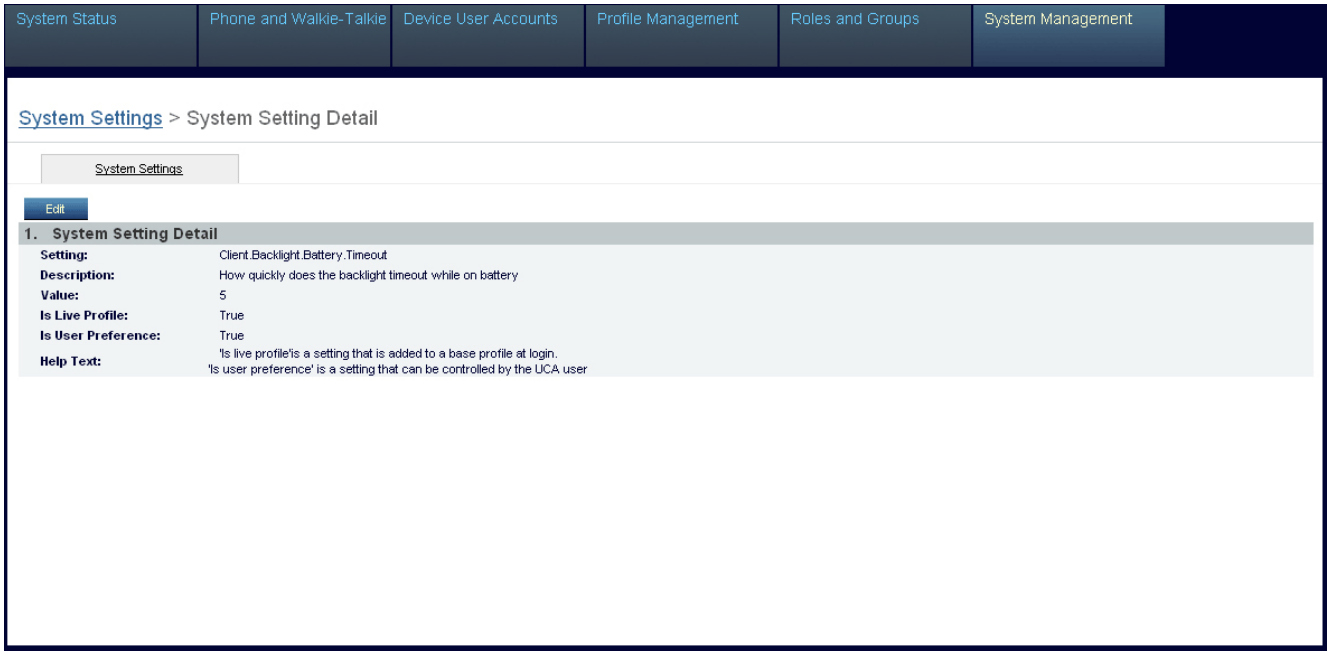

System Status
Phone and Walkie-Talkie
Device User Accounts
Profile Management
Roles and Groups
System Management
System Settings > System Setting Detail
System Settings
Edit
1. System Setting Detail
Setting: Client.Backlight.Battery.Timeout
Description: How quickly does the backlight timeout while on battery
Value: 5
Is Live Profile: True
Is User Preference: True
Help Text: 'Is live profile'is a setting that is added to a base profile at login.
'Is user preference' is a setting that can be controlled by the UCA user

Figure 9-39 *Client Settings View*

Phone Settings

Phone settings are used for PBX and phone system settings used with the UCA server.

Settings like Phone Gateway IP, Port, Type, etc. can also be viewed and configured.

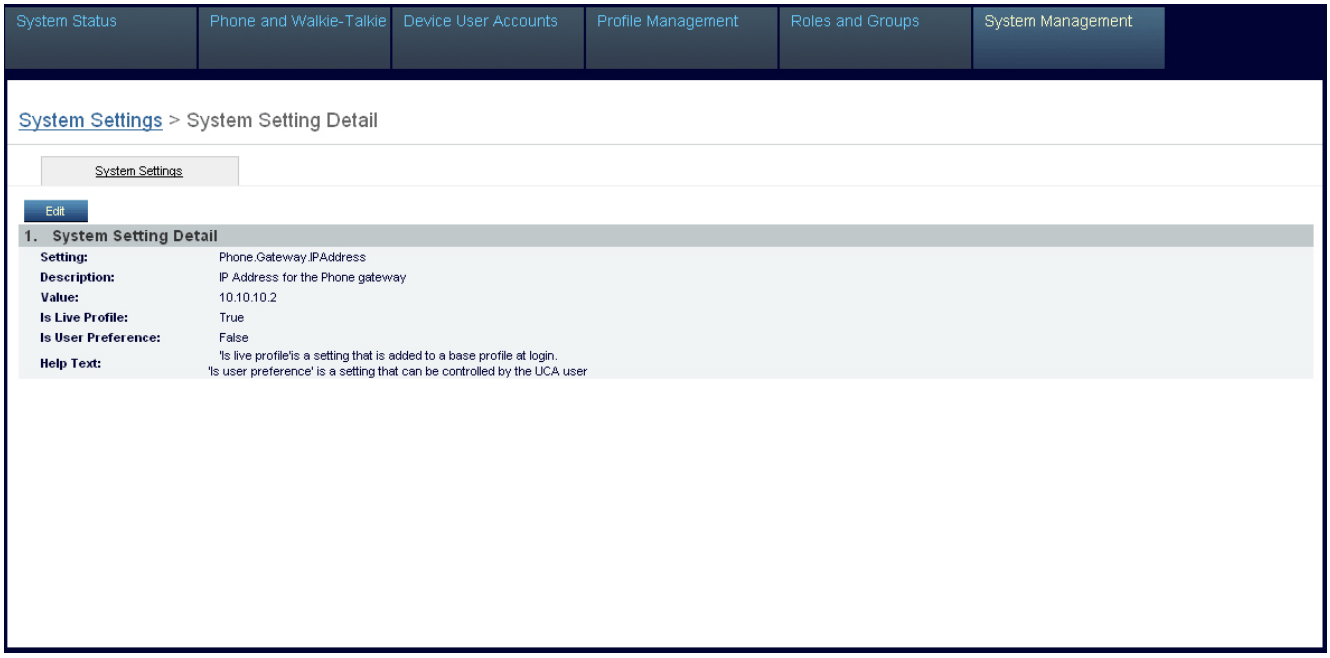

System Status
Phone and Walkie-Talkie
Device User Accounts
Profile Management
Roles and Groups
System Management
System Settings > System Setting Detail
System Settings
Edit
1. System Setting Detail
Setting: Phone.Gateway.IPAddress
Description: IP Address for the Phone gateway
Value: 10.10.10.2
Is Live Profile: True
Is User Preference: False
Help Text: 'Is live profile'is a setting that is added to a base profile at login. 'Is user preference' is a setting that can be controlled by the UCA user

Figure 9-40 *Phone Settings View*

Phonebook Settings

Phonebook settings are values used to configure the way users view phone book extensions. Phonebook settings also allow large phone extension lists to be broken out into specified range values.

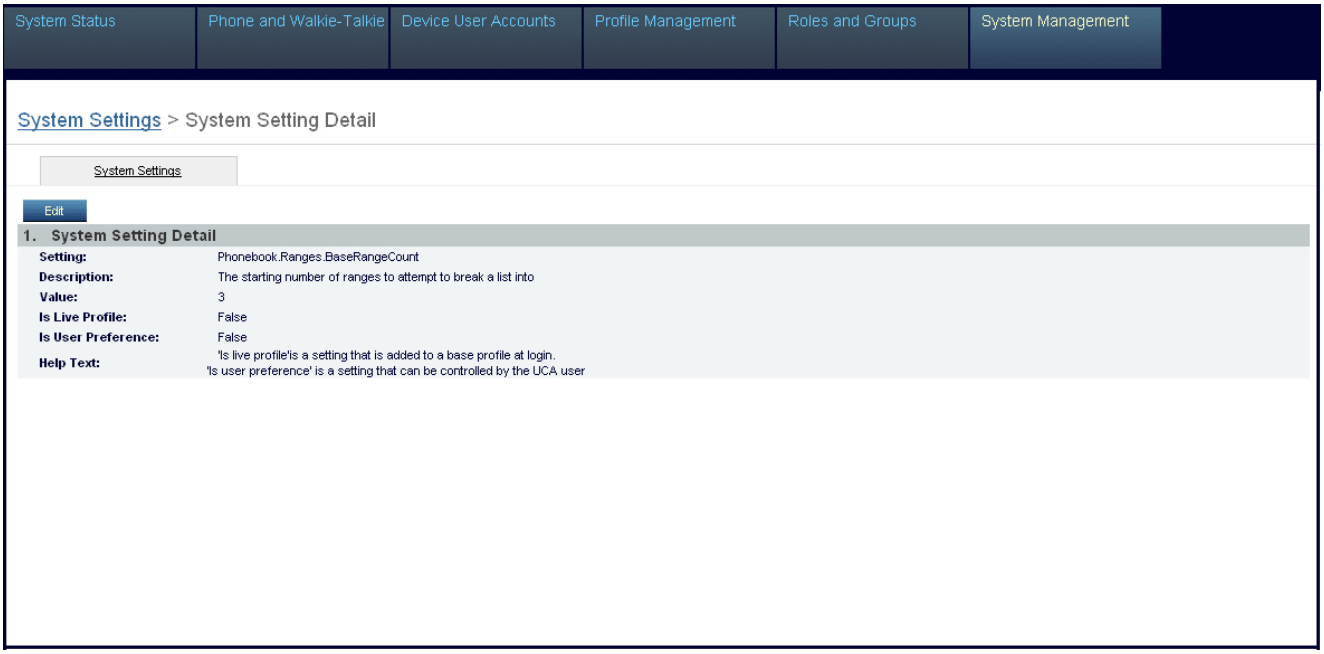

System Status
Phone and Walkie-Talkie
Device User Accounts
Profile Management
Roles and Groups
System Management
System Settings > System Setting Detail
System Settings
Edit
1. System Setting Detail
Setting: Phonebook.Ranges.BaseRangeCount
Description: The starting number of ranges to attempt to break a list into
Value: 3
Is Live Profile: False
Is User Preference: False
Help Text: 'Is live profile'is a setting that is added to a base profile at login. 'Is user preference' is a setting that can be controlled by the UCA user

Figure 9-41 *Phonebook Settings View*

High Level System Settings

High level UCA server operational settings can be modified by a UCA Administrator who can view and set many system options for Profiles, IP addresses, Logins, Timeouts, Web Page allocation, Database Maintenance and Media Relay values.

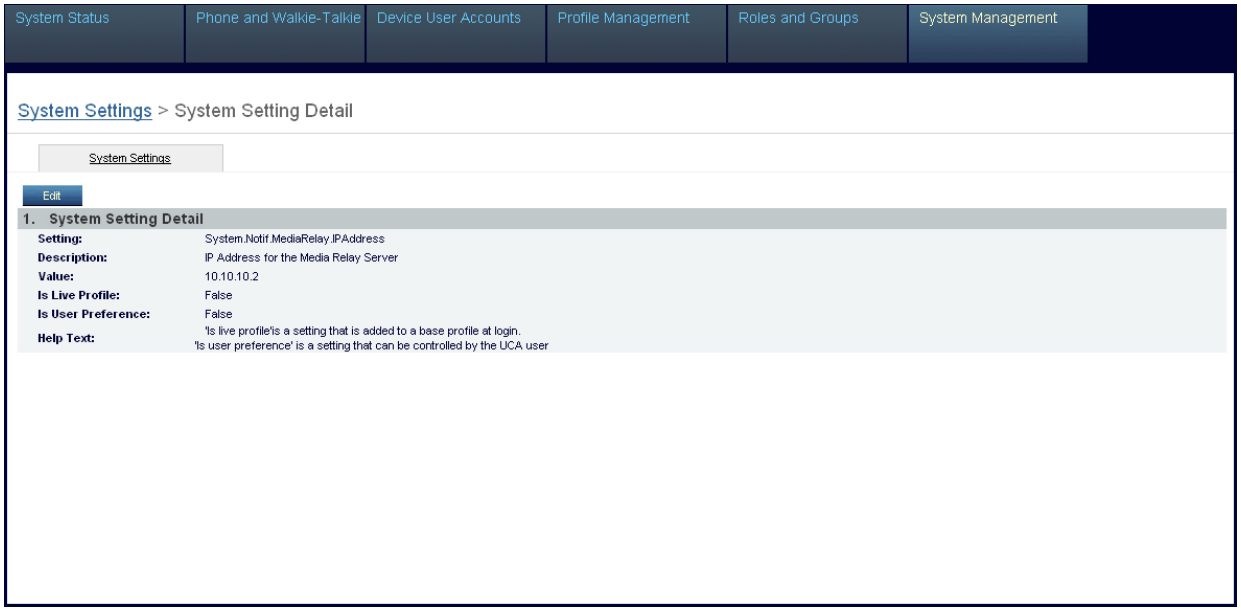

System Status
Phone and Walkie-Talkie
Device User Accounts
Profile Management
Roles and Groups
System Management
System Settings > System Setting Detail
System Settings
Edit
1. System Setting Detail
Setting: System.Notif.MediaRelay.IPAddress
Description: IP Address for the Media Relay Server
Value: 10.10.10.2
Is Live Profile: False
Is User Preference: False
Help Text: 'Is live profile'is a setting that is added to a base profile at login.
'Is user preference' is a setting that can be controlled by the UCA user

Figure 9-42 *System Settings View*

Configure System Settings

To configure System Settings settings:

1. Click the *System Management* tab > *System Settings* tab on the UCA server WebConsole GUI window to display the tab view.

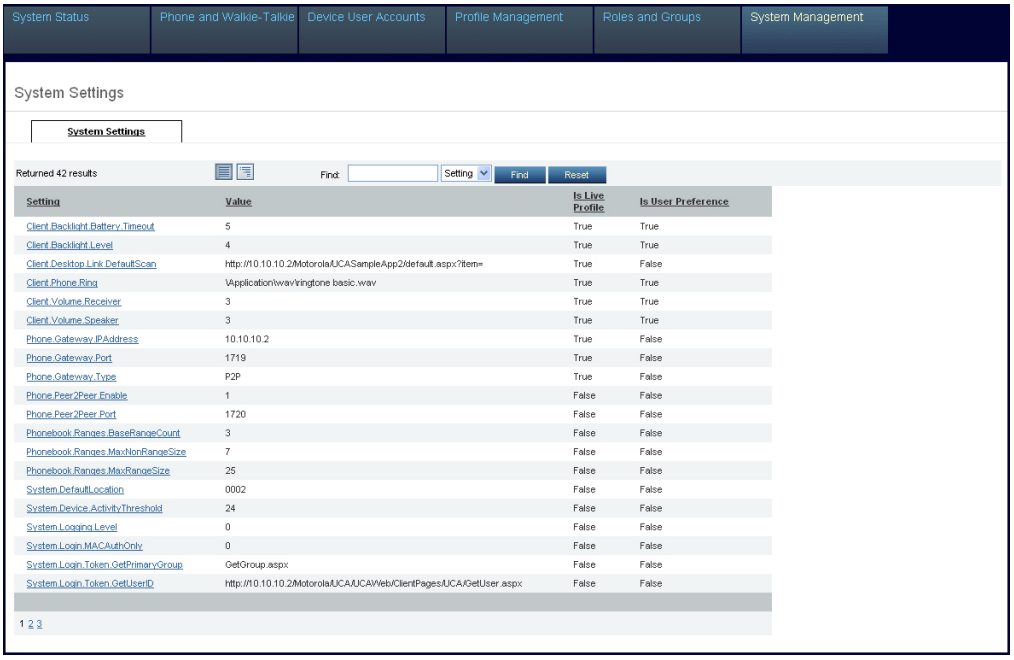


Figure 9-43 *System Management View*

2. Select an entry to configure in the *Settings* column.
3. In the detail view of the setting entry, click **Edit**.
4. Modify the setting, as needed.
5. Click **Save** or **Cancel** to terminate the session.

NOTE See *System Settings on page 6-48* for a detailed list of the system settings for CA50 Client/Server software.

Console Users

Console Users provides the ability to create, edit and delete UCA System users.

The *Console Users* view displays a user's *UserID*, First name, Last Name, Status and dates for account creation and activity.

To view Console Users:

- Click the *Console Users* tab on the UCA server WebConsole GUI window to display the tab view.

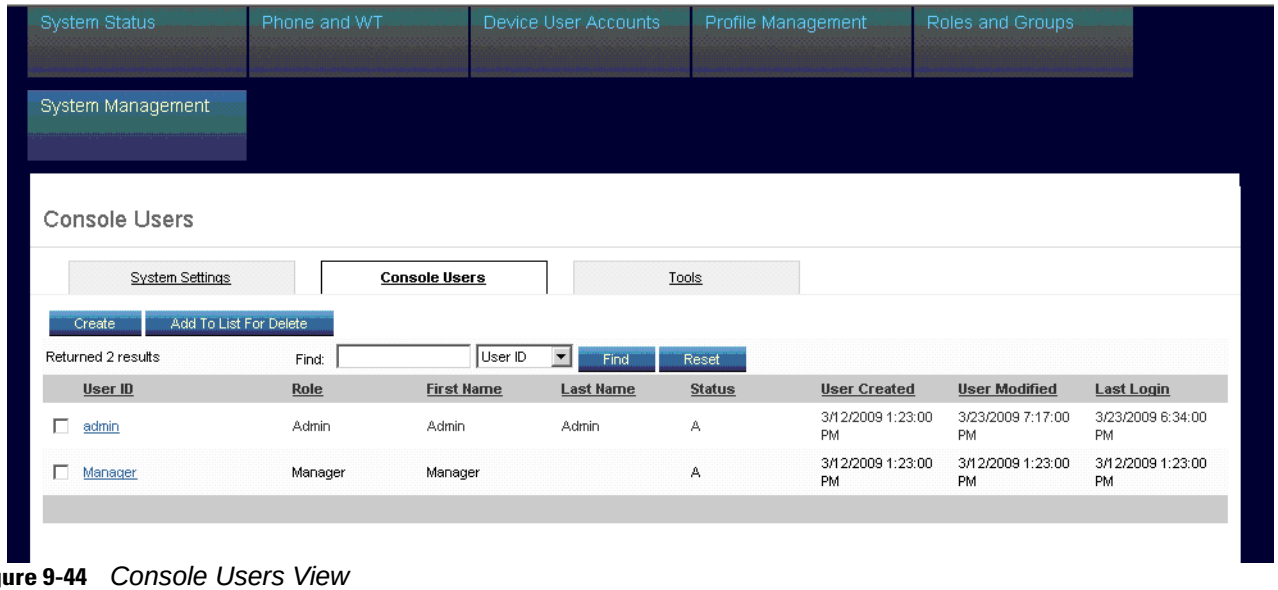


Figure 9-44 *Console Users View*

Creating a New Console User

To create a console user in the UCA server:

1. Click the *System Management* tab > **Console Users** tab > **Create**.
   The *Create User* window displays.

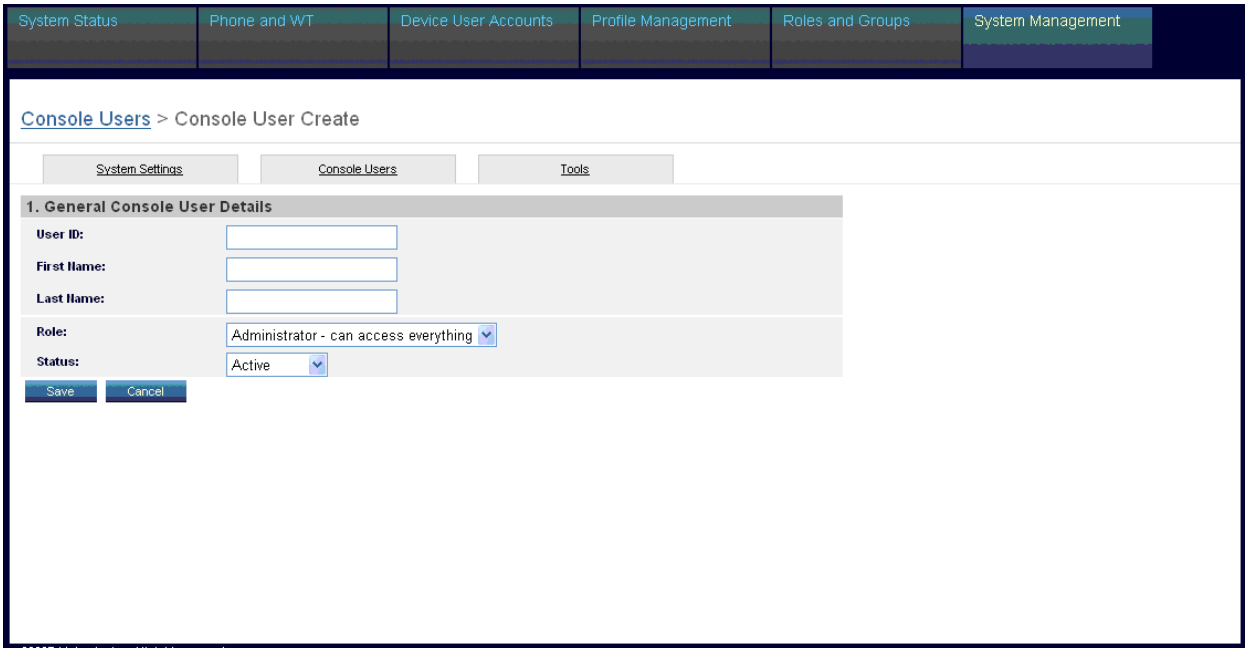


Figure 9-45 *Create User Window*

2. Enter in the user's User ID, first and last name in the appropriate text boxes.
3. Select a role from the drop down list menu.
4. Select a status from the drop down list menu.
5. Click **Save** to store the user
   or
   Click **Cancel** to terminate the session.

NOTE The UCA server supports two roles: Administrator and Manager. Administrator has full permissions and can perform all actions available through the UCA Console. Manager has limited permissions and can only view *System Status* and add, edit and remove CA50 device users.

Editing Console User Information

To edit console user information in the UCA server:

1. Click the *System Management* tab > **Console Users** tab.
2. Click the *User ID* to edit from the list of users in the User ID column.
   The Console User *Detail* window displays.

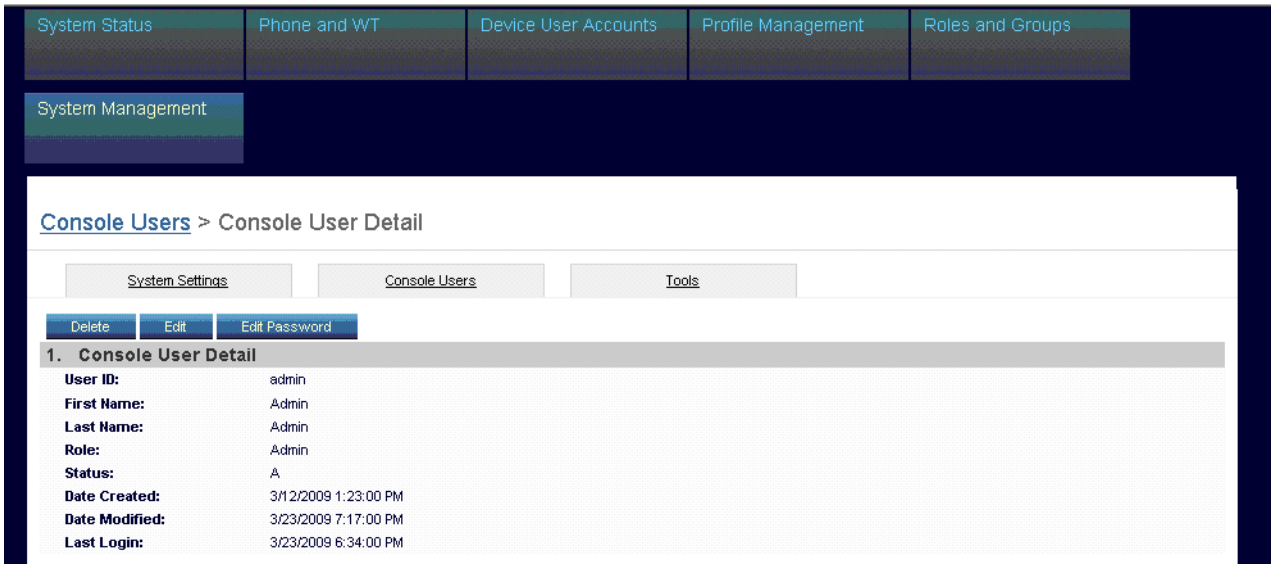


Figure 9-46 *Console User Detail Window*

3. Click **Edit**.
   The Edit Console User window displays.

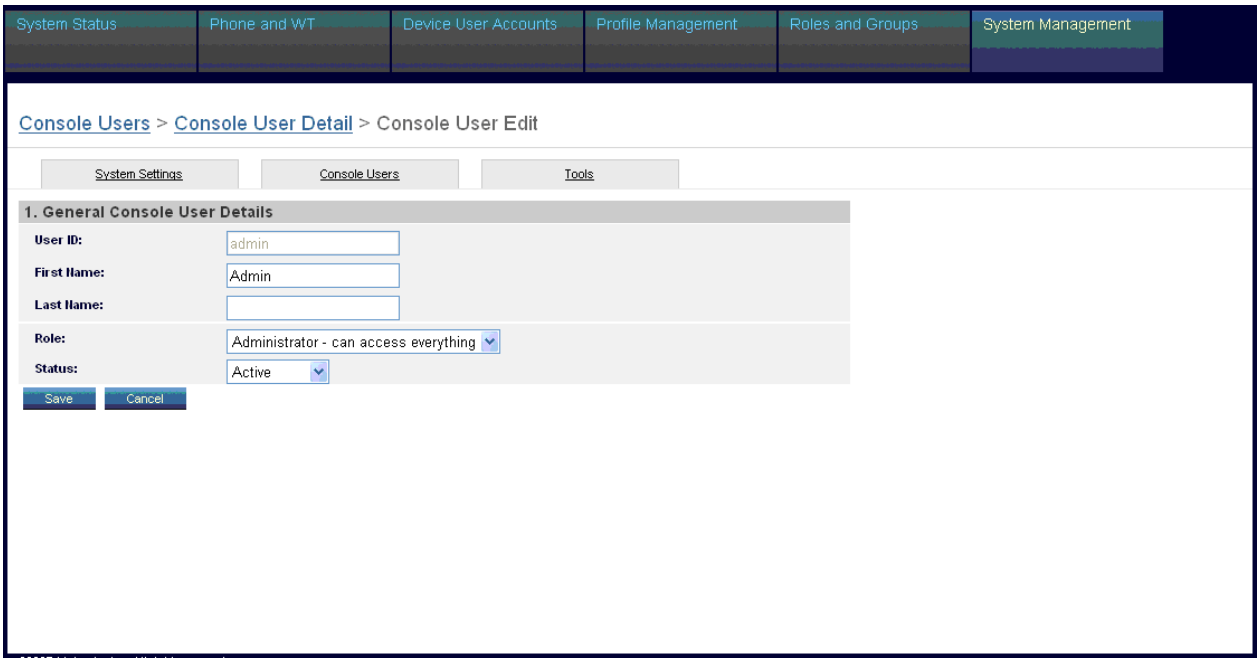


Figure 9-47 *Edit User Window*

4. Enter the console user information to change in the appropriate text boxes.

5. Click **Save** to save the edits
   or
   Click **Cancel** to terminate the session.

Editing a Console User Password

To edit a console user's password in the UCA server:

1. Click the *System Management* tab > *Console Users* tab.
2. Click the *User Id* to edit from the list of console users in the *User ID* column.
   The *Console User Detail* window displays.

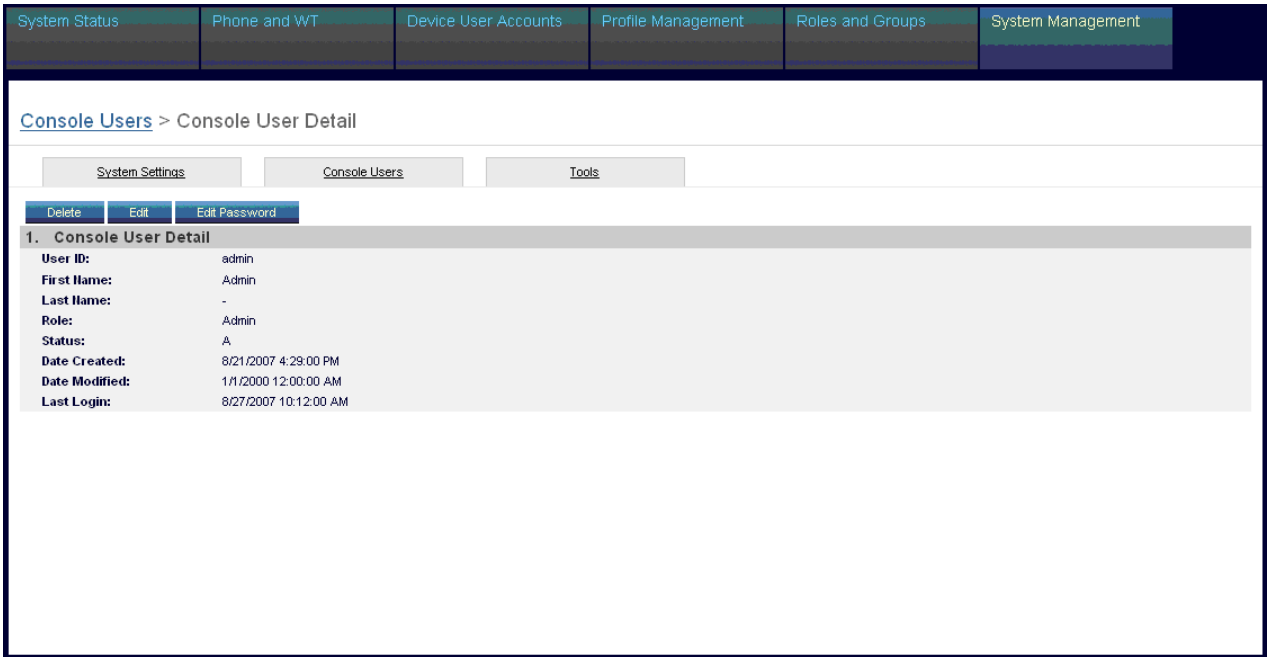


Figure 9-48 *Console User Detail Window*

3. Click **Edit Password**.
   The Edit Console User Password window displays.

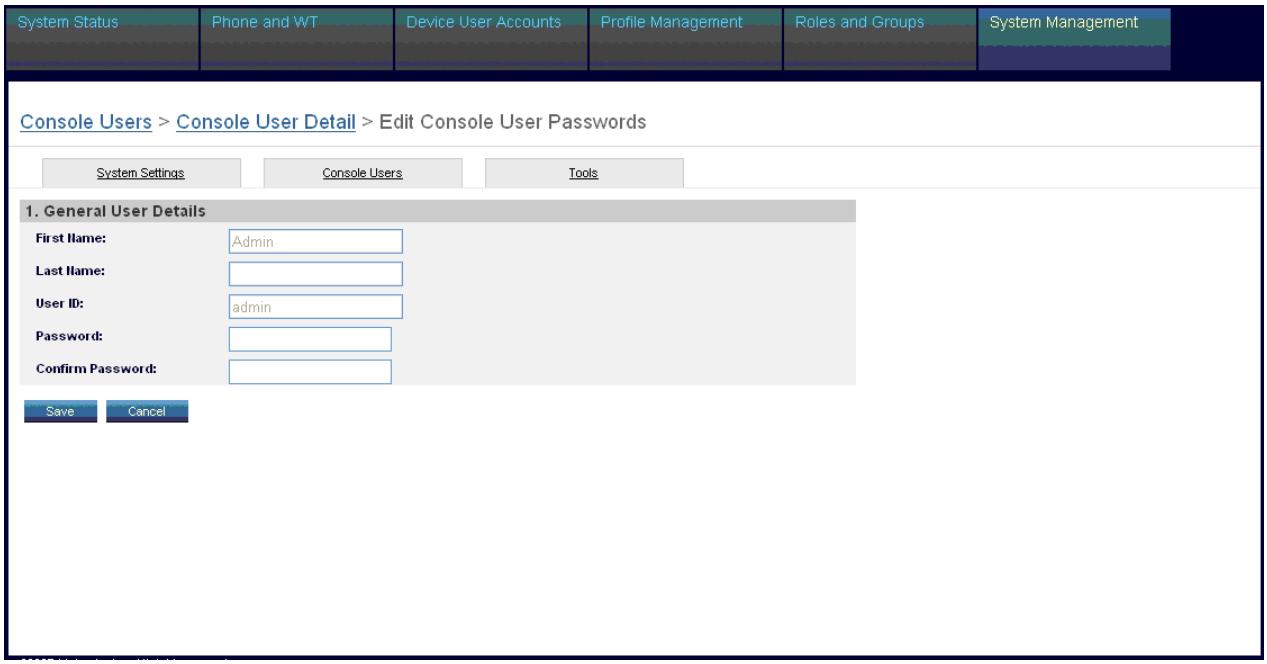


Figure 9-49 *Edit Console User Passwords Window*

4. Enter the new password into the *Password* text box.
5. Confirm the new password in the *Confirm Password* text box.
6. Click **Save** to change the password
   or
   Click **Cancel** to terminate the session.

Deleting a Console User

See *Delete Utility on page 9-52* for information about deleting entries in the UCA server.

Tools

Tools present two options to a console user.

- MSP Staging: This is a link to MSP Stage. After setting the system setting *System.Tools.Stage* to point to MSP stage in your environment, this link can be used for quick access to the Motorola MSP Stage product.
- Debug Trace Log Utilities: Through the *Tools* tab the console user can export, purge, and export/purge the debug trace log found in the UCA database. The trace logs are exported to the directory specified in the *System Setting System.Maintenance.DB.Export.Path*. The trace files are exported in CSV format for easy loading into Microsoft® Excel.

To access *Tools*:

1. Click the *System Management* tab > *Tools* tab.
   The *Tools* window displays.

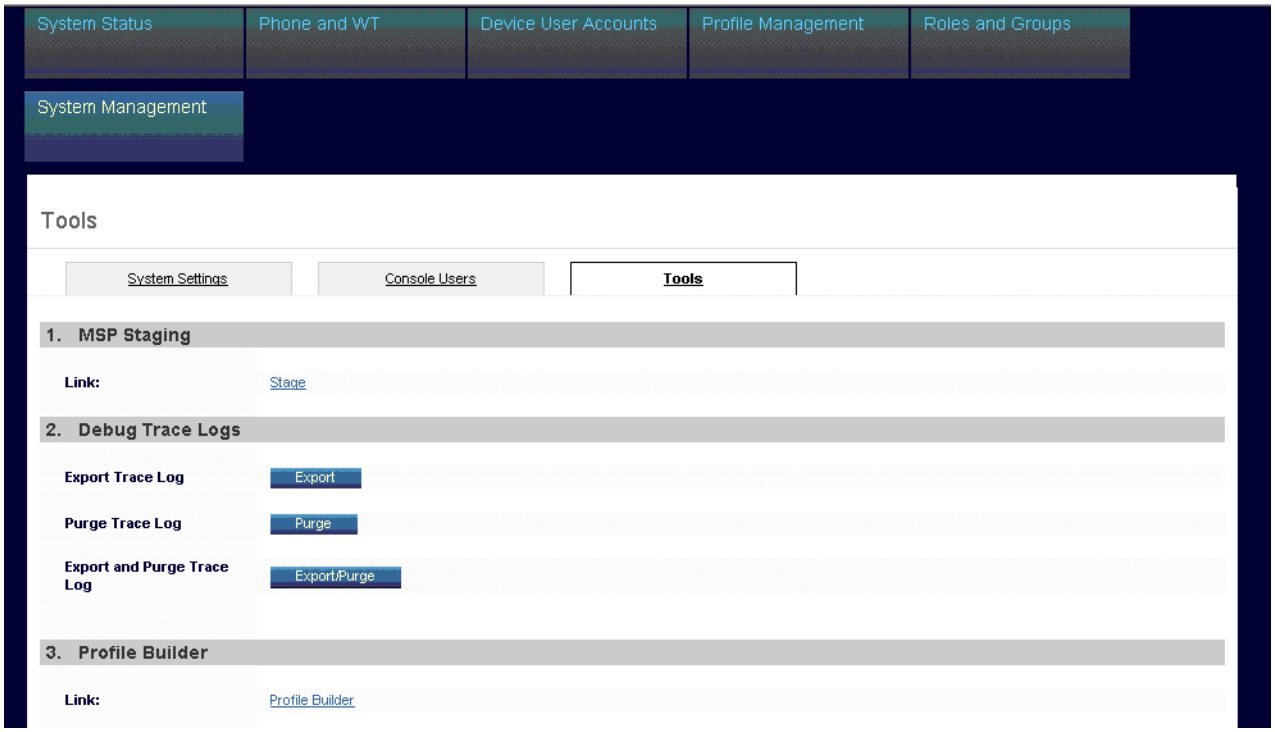


Figure 9-50 *Tools Window*

Profile builder is installed on the UCA server by default. This link is provided so that Profile builder can be installed on another system. The provided link points to the profile builder installable and not the profile builder itself. After installing the profile builder, launch the profile builder from:

Start > All Programs>Motorola>UCA Server>Utils>Profile Builder

Exporting the Debug Trace Log

To export the debug trace logs found in the UCA server:

1. Click the *System Management* tab > *Tools* tab > **Export**.
   The *Export Debug Trace Log* window displays.

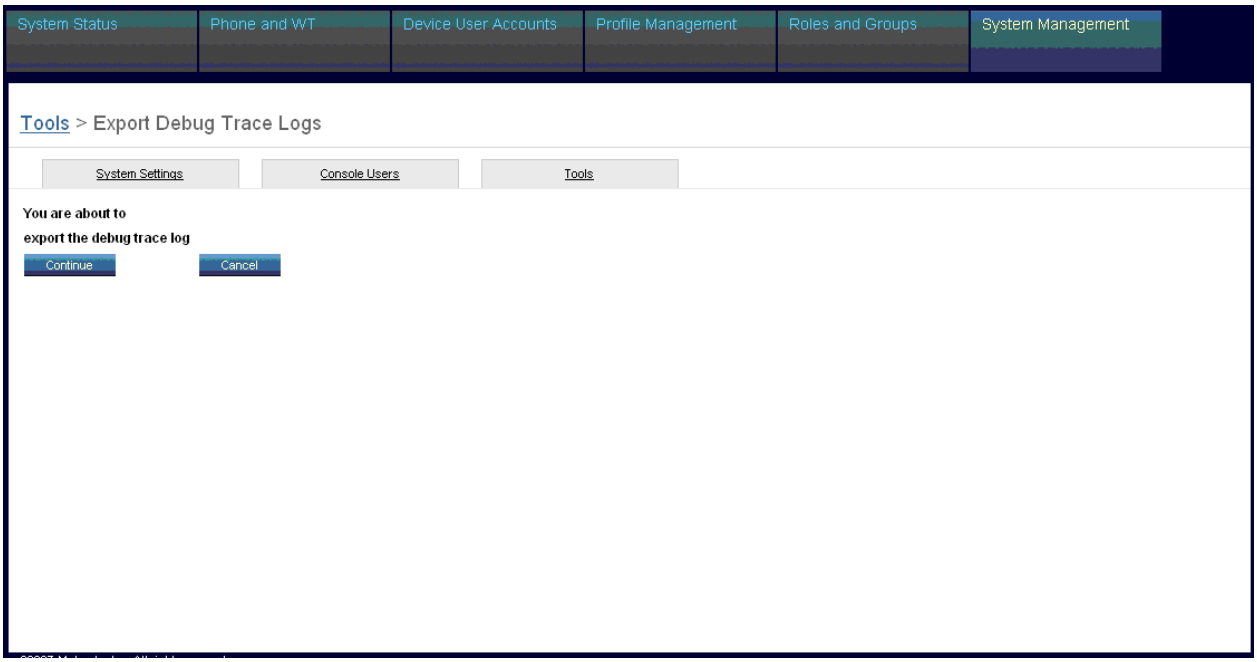


Figure 9-51 *Export Debug Trace Log Window*

2. Select **Continue** to export the trace log.
3. Select **Cancel** to return to the *Tools* tab.

Purging the Debug Trace Log

To purge the debug trace logs found in the UCA server:

1. Click the *System Management* tab > *Tools* tab > **Purge**.
   The *Purge Debug Trace Log* window displays.

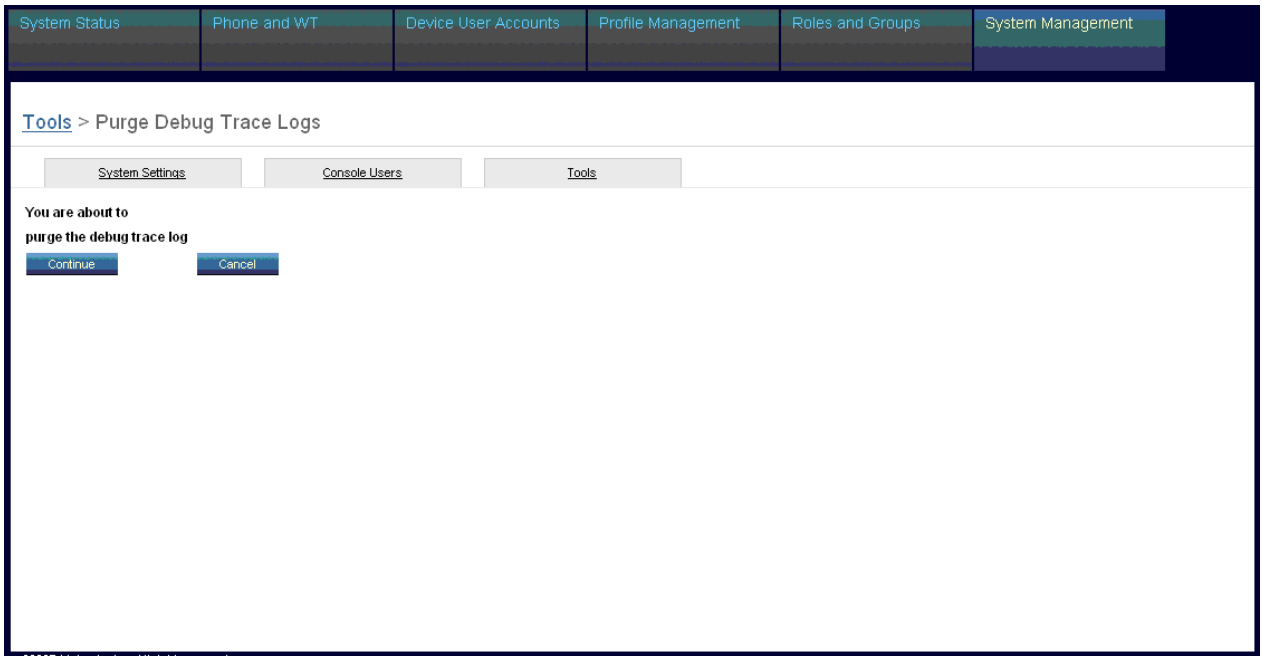


Figure 9-52 *Purge Debug Trace Log Window*

2. Select **Continue** to purge the trace log.
3. Select **Cancel** to return to the *Tools* tab.

Exporting and Purging the Debug Trace Log

To export and purge the debug trace logs found in the UCA server:

1. Click the *System Management* tab > *Tools* tab > *Export/Purge*.
   The *Export/Purge Debug Trace Log* window displays.

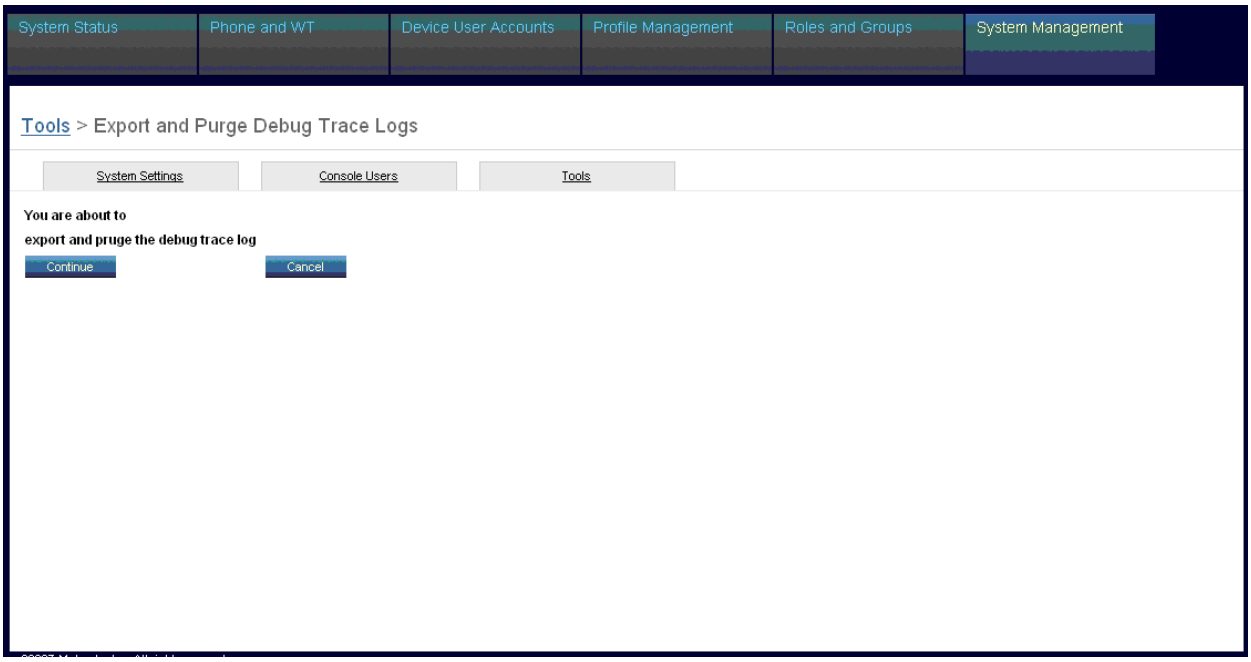


Figure 9-53 *Export/Purge Debug Trace Log Window*

2. Select **Continue** to export and purge the trace log.
3. Select **Cancel** to return to the *Tools* tab.

UCA Server WebConsole GUI Utilities

Find Utility

All tab views on the UCA server WebConsole GUI, except *System Status*, provide a *Find Utility* to help locate specific entries.

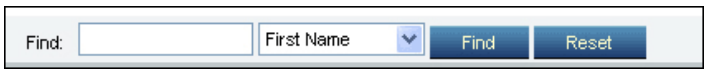


Figure 9-54 *Find Utility Tool Bar*

To find an entry on a tab view:

1. Type an entry to find in the *Find:* text box.
2. Select an option from the pull down menu. This selection identifies the what type of item was entered in step 1 above (e.g., First Name, Last Name, Phone Extension, etc.).
3. Click **Find** to begin the search.
4. Search results display in the current window.

Click **Reset** to clear the text box.

Delete Utility

The delete utility allows entries in the UCA server database to be deleted on all main tab views in the UCA server WebConsole GUI, except *System Status*. The utility provides two methods for deleting table entries in the UCA server database:

- Delete one entry at a time
- Delete multiple entries.

Deleting a Single Entry

This method allows a single deletion of an entry in a view.

To delete a single entry on a tab view:

1. Click the appropriate tab on the WebConsole GUI window to display the view.
2. Select the item in the list to delete by clicking on the item ID or the underlined item in the first column. The item *Detail* window displays.

3. On the *Detail* window, click **Delete**.

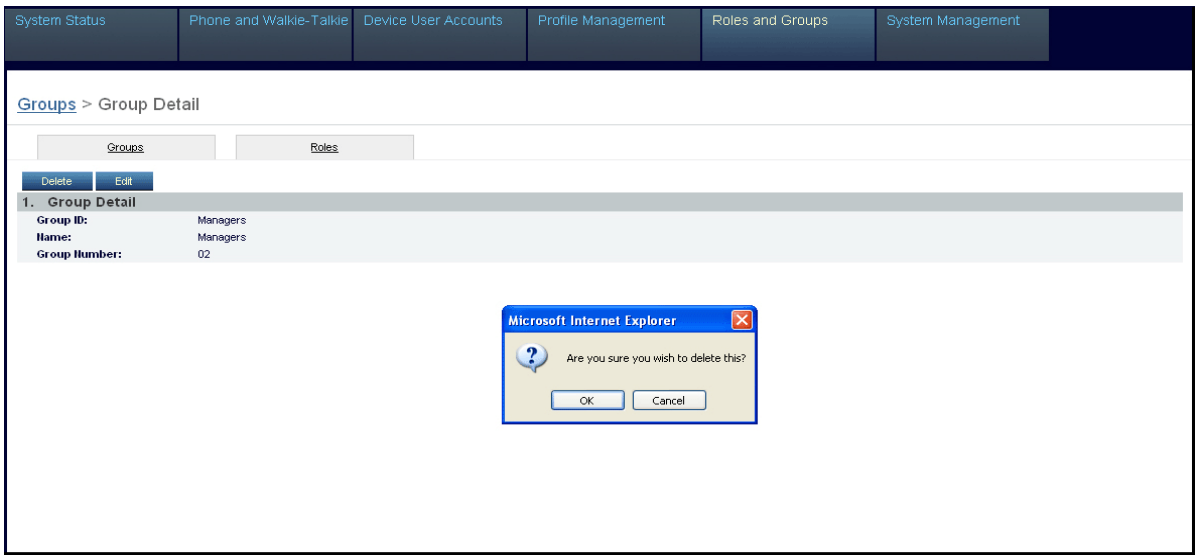


Figure 9-55 *Detail Window*

4. Click **OK** in the *Are you sure you want to delete this?* message box to delete the entry from the database.

Deleting Multiple Entries

This method includes an *Add to List for Delete Utility* that allows entries in a tab view to be added to a list of entries to delete.

To delete multiple entries on a tab view:

1. Click the appropriate tab on the WebConsole GUI window to display the view.

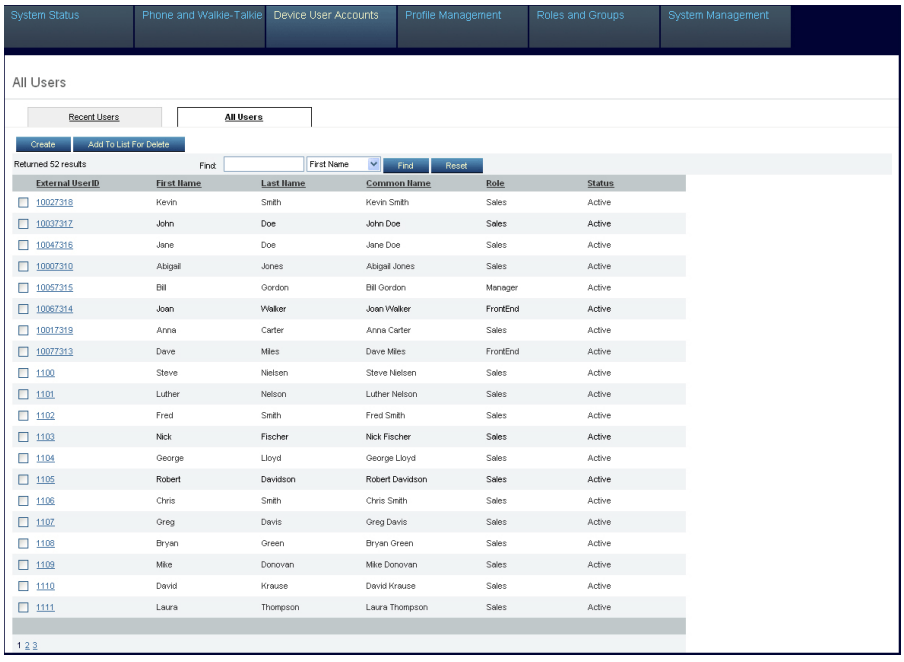

System Status
Phone and Walkie-Talkie
Device User Accounts
Profile Management
Roles and Groups
System Management
All Users
Recent Users
All Users
Create
Add To List For Delete
Returned 52 results
Find:
First Name
Find
Reset
External UserID
First Name
Last Name
Common Name
Role
Status
10027318 Kevin Smith Kevin Smith Sales Active
10037317 John Doe John Doe Sales Active
10047316 Jane Doe Jane Doe Sales Active
10007310 Abigail Jones Abigail Jones Sales Active
10057315 Bill Gordon Bill Gordon Manager Active
10067314 Joan Walker Joan Walker FrontEnd Active
10017319 Anna Carter Anna Carter Sales Active
10077313 Dave Miles Dave Miles FrontEnd Active
1100 Steve Nielsen Steve Nielsen Sales Active
1101 Luther Nelson Luther Nelson Sales Active
1102 Fred Smith Fred Smith Sales Active
1103 Nick Fischer Nick Fischer Sales Active
1104 George Lloyd George Lloyd Sales Active
1105 Robert Davidson Robert Davidson Sales Active
1106 Chris Smith Chris Smith Sales Active
1107 Greg Davis Greg Davis Sales Active
1108 Bryan Green Bryan Green Sales Active
1109 Mike Donovan Mike Donovan Sales Active
1110 David Krause David Krause Sales Active
1111 Laura Thompson Laura Thompson Sales Active
1 2 3

Figure 9-56 *All Users View*

2. Click the check boxes to the left of each entry to delete.

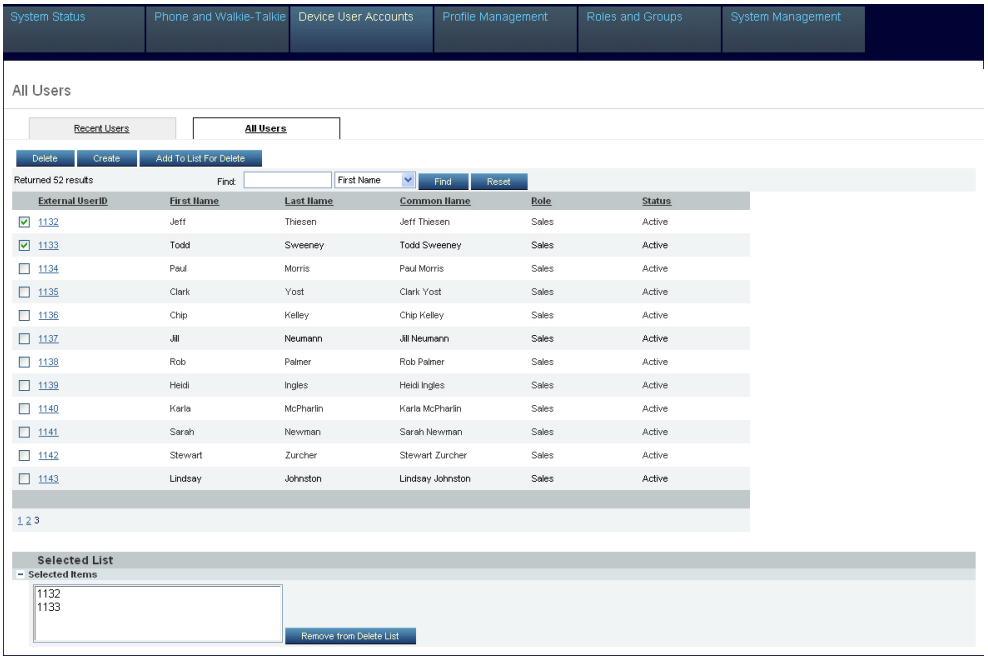


Figure 9-57 *Selections to Delete*

3. Click **Add to List For Delete**.
4. Each entry checked for deletion displays in the *Selected Item List* box on the bottom of the tab view.

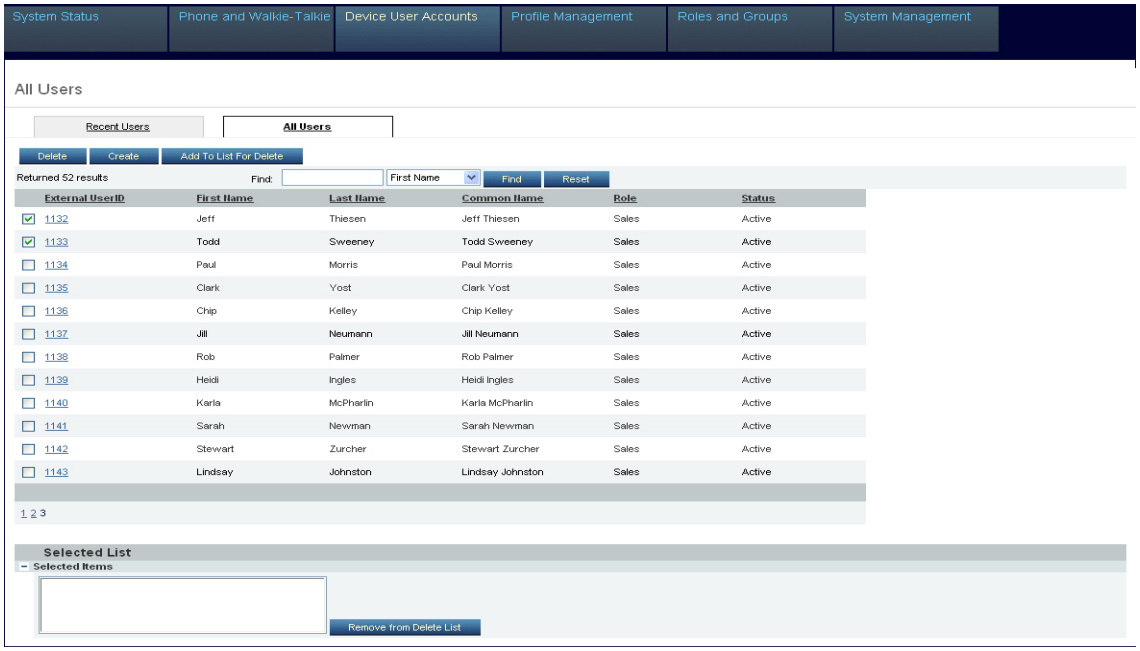


Figure 9-58 *Entries to Delete*

NOTE To remove an entry from the selected delete list, click the entry in the *Selected Item List* box and click **Remove from Delete List**.

5. Click **Delete** to delete all selected entries.
6. A confirmation dialogue displays to verify the delete.
7. Click **Yes** to delete the selected entries.

Import Utility

The Import Utility allows some of the lists to be imported via CSV files instead of manually entering the data.

To import from a CSV file:

1. On the *Device User Accounts* tab, click **Import**.

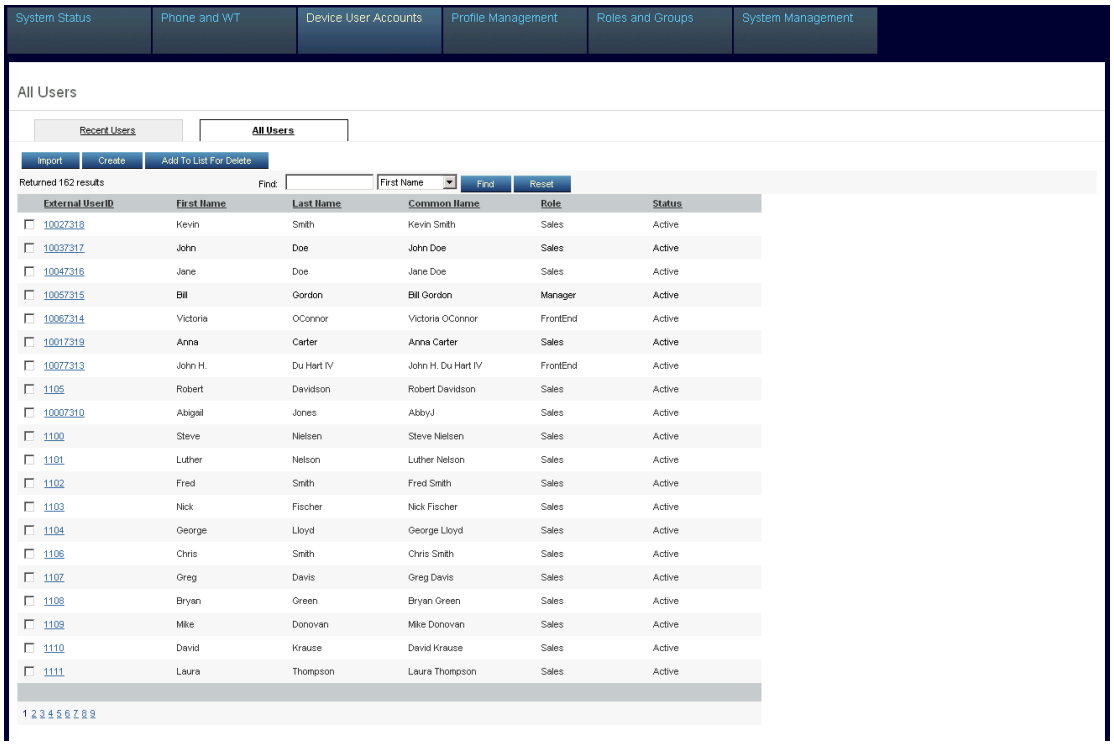

System Status
Phone and WT
Device User Accounts
Profile Management
Roles and Groups
System Management
All Users
Recent Users
All Users
Import
Create
Add To List For Delete
Returned 162 results
Find:
First Name
Find
Reset
External UserID
First Name
Last Name
Common Name
Role
Status
10027318 Kevin Smith Kevin Smith Sales Active
10037317 John Doe John Doe Sales Active
10047316 Jane Doe Jane Doe Sales Active
10057315 Bill Gordon Bill Gordon Manager Active
10067314 Victoria OConnor Victoria OConnor FrontEnd Active
10017319 Anna Carter Anna Carter Sales Active
10077313 John H. Du Hart IV John H. Du Hart IV FrontEnd Active
1105 Robert Davidson Robert Davidson Sales Active
10007310 Abigail Jones AbbyJ Sales Active
1100 Steve Nielsen Steve Nielsen Sales Active
1101 Luther Nelson Luther Nelson Sales Active
1102 Fred Smith Fred Smith Sales Active
1103 Nick Fischer Nick Fischer Sales Active
1104 George Lloyd George Lloyd Sales Active
1106 Chris Smith Chris Smith Sales Active
1107 Greg Davis Greg Davis Sales Active
1108 Bryan Green Bryan Green Sales Active
1109 Mike Donovan Mike Donovan Sales Active
1110 David Krause David Krause Sales Active
1111 Laura Thompson Laura Thompson Sales Active
1 2 3 4 5 6 7 8 9

Figure 9-59 *Device User Accounts tab*

2. The *Import* screen displays.

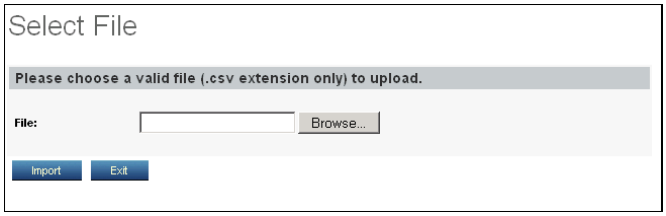


Figure 9-60

3. Click **Browse** and use the *Choose file* dialog to select the CSV file to import.
4. Click **Import** to initiate the import process.
5. A results screen displays to summarize the import.
6. Click **Back** to select a different file or **Exit** to go back to the list page.

(Page intentionally blank.)

Chapter 10 CA50 Business Application Development

Overview

This chapter provides guidance for developing Web applications that reside on a Web server and are accessed by the CA50 UCA client application.

Valuable topics in this chapter include:

- Business application overview (*page 10-2*)
- CA50 Web application developer guide (*page 10-3*)
- Launching a business application (*page 10-3*)
- Automating the launch of a business application (*page 10-3*)
- Employee identity validation (*page 10-3*)
- Request format (*page 10-4*)
- Response format (*page 10-4*)
- HTML elements (*page 10-5*)
- Best practices and considerations for CA50 development (*page 10-12*)
- Data collection (*page 10-12*)
- Access to communication options and the Web application (*page 10-12*)
- Business application extensions Web service (*page 10-16*).

NOTE For additional sample business applications see *Appendix I, Business Applications Supplement*.

Business Application Overview

Business applications are third party developed, custom applications for use with a CA50. A business application can control the following CA50 features and functions:

- scanner
- display
- LEDs
- play audio files
- vibrator motor

Third party business applications cannot run directly on the CA50. Instead they are developed for, hosted by and run on the UCA server or another addressable Web server in the enterprise. The UCA client application allows a user access to server based third party business applications via an HTTP 1.1 compliant Web browser. A user can interact with the business application via the CA Menu option which accesses the Web browser. Business applications can be hosted on the UCA server or other addressable Web servers in the enterprise.

Multiple business applications can be accessed by a CA50. From the menu, a user selects one of the business applications for operation. Only one business application can be run at a time on the CA50.

For example, in a price lookup business application, the CA50 device scans and decodes bar code information and transmits to the UCA server for processing. The server information associated with the bar code is then sent to the CA50 for display and use by the store associate. Transmission of the data between the CA50 and UCA server occurs over the wireless infrastructure of the facility.

NOTE After loading the UCA server software, sample CA50 business applications can be found in the following directory: \Program Files\Motorola\UCA Server\Development Toolkit\.

NOTE 1. For a complete checklist outlining all the development and deployment activities required to implement a CA50 solution, see the *Development and Deployment Checklist on page 2-1*.

2. For a table listing the most commonly requested topics within this manual, see the Quick Startup Guide on the last page of the manual.

Business Application Development Environment

Developers familiar with standard Web site development methodologies are capable of business application development for the CA50. Business applications for the CA50 can be thought of as Web sites. Any server side programming languages that are used in standard Web site development, such as C#, JSP, etc., can be used to develop CA50 business applications. The only requirement is that the output generated by the server (HTML tags) and sent to the CA50 must be compatible with the tags supported by the UCA client application. For additional information on the UCA client application see *UCA Client Application Overview on page 5-1*.

CA50 Web Application Developer Guide

This section provides a functional description of the operation of the browser and the customized tags.

To work with the UCA client application browser, the responses generated by the Web server adhere to the HTML specification to the greatest extent possible. Due to the small display on the CA50 client, many HTML tags are not applicable. For example, a graphical submit button would use most of the rendering window for the browser. In addition, the UCA client application does not have a pointing device and has limited physical buttons or keys. Therefore, proprietary HTML element attributes were designed and implemented to submit form data based on scanner data or physical buttons as input tag. There are also some proprietary HTML meta tags that provide additional functionality.

The UCA server installation includes a few sample CA50 Web applications. In the default installation, the source can be found in the following directories:

\Program FIles\Motorola\UCA Server\Development Toolkit\Sample Lookup and
\Program FIles\Motorola\UCA Server\Development Toolkit\Sample Lookup2.

Launching a Business Application

Business applications can be launched by associating the application URL to the selection of a menu item in the base profile, or by associating the application URL to an action taken from the desktop (like scanning a bar code).

In the default sample profiles installed on the server, see the desktop element and the application menu elements for examples of both. These examples can be modified to reference the URL of any CA50 Web application. For more details, see *Chapter 7, XML Profile for Configuration*.

Automating the Launch of the UCA Client Application

For details, see *page 12-2*.

Employee Identity Validation

For details, see *PIN Input*, *Figure 4-5 on page 4-3*.

Request Format

The first line of the Request is the "Request Line". A "Request Line" consists of the request method (GET or POST), the absolute URI being requested and the HTTP version. The "Request Line" is terminated by a CRLF. Requests use the GET method when form data is not being submitted to the server. The GET or the POST method is used to submit form data. The "method" element of the form selector decides whether or not the form data uses the GET or POST method. The UCA client application browser uses the POST method when the "method" element is not contained in the form selector. The "Request Line" is followed by several header fields. Header fields have the format "name:value" pair. The following examples show the initial connection Request and a typical form submission Request:

```
HTTP Command:GET
URI:
/Server/GetUser.apsx?sStoreNo=0002&sIPAddress=111.111.111.1111&sSerialNo=00000000&sMAC=11AA1314151
6&sDeviceType=Motorola UCA&sClientVersion= 1.01 HTTP/1.1<CF><LF>
User-Agent: UCAClient
Host:       111.111.111.111<CR><LF>
Connection: Keep-Alive<CR><LF><CR><LF>
```

NOTE The *SerialNo* name value pair is not fully implemented at this time and the SerialNo value is invalid.

```
HTTP Command:       POST
URI:                /Server/GetUser.apsx
HTTP Version        HTTP/1.1<CR><LF>
Accept:             */*
Content-Type:       application/x-www-form-uriencoded.charset=utf-8<CR><LF>
IPAddress:          111.111.111.111<CR><LF>
SerialNo:           12345678<CR><LF>
MAC:                11AA13141516<CR><LF>
PlatformInfo:       Motorola WinCE<CR><LF>
Host:               111.111.111.111<CR><LF>
User-Agent:         htmlayout 3.1 ; UCAClient ; Windows CE<CR><LF>
Content-Length:     84
Connection:         Keep-Alive<CR:<LF>
Cache-Control:      no-cache<CR><LF>
Form Data:          username=10007310
```

NOTE The *SerialNo* name value pair is not fully implemented at this time and the SerialNo value is invalid.

Motorola's additional HTTP headers provided by the UCA client application browser are defined as follows.

IPAddress - IP address of the device.

SerialNo - As mentioned earlier, this name value pair is currently not fully implemented at this time and the SerialNo is invalid.

MAC - The MAC address of the device

PlatformInfo - This is the platform information obtained from the Windows CE API SystemParametersInfo function.

Response Format

The Web server is responsible for the content of the Response header. The Response body must be a well-formed HTML document for the UCA client application browser to parse.

HTML Elements

The UCA client application browser provides rendering of HTML Web pages for both standard HTML elements and proprietary HTML elements.

Virtual (non-visible) buttons are used to provide a way for the UCA client application browser to send event data such as function key presses, scanner data, etc. to the server to be processed.

Given the limited user interface requirements and the display size, only a small subset of HTML selectors (tags) and attributes are applicable. The UCAClient's physical display size is 64 x 96 pixels. The intent is that the server sends preformatted text, inserting spaces as needed, to make columns line up.

Standard HTML Elements

As mentioned earlier, given the display size of the CA50 device, most standard HTML elements are not applicable for business applications being run by the CA50 device. For the most part, graphical controls (buttons, drop down lists, scrolling lists, visible input fields, etc.), hyperlinks, frames, tables, style sheets, and formatting tags should NOT be used. Also, JavaScript is NOT supported.

Proprietary HTML Elements

Motorola developed proprietary HTML elements to be used in business applications. The elements in *Table 10-1* (below) are within the <head> element; the elements in *Table 10-2 on page 10-8* are within the <body> section.

HTML Elements Contained in the Head Element

Table 10-1 *HTML Elements Contained in the Head Element*

Element	Description
`<meta http-equiv = "ExitBusApp" content="yes"/>`	This meta tag causes the business application to exit and the UCA client application goes back to the UCA client application desktop. This means the user is still logged in and has access to the menus, walkie-talkie or phone calls.
`<meta http-equiv ="scanner" content="[parameter]">`	Parameters: *enabled-* Enables the scanner. *disabled-* Disables the scanner. *autoenter-* Automatically appends a carriage return to the end of any bar codes scanned. autotab- automatically appends a tab to the end of any bar codes scanned. This meta tag allows the business application developer to enable or disable the scanner as well as append a carriage return or tab to the end of the scanned label. **Note:** The enable and disable parameters are only necessary when the business application developer wishes to use the custom "scannernavigate" meta tag.
`<meta http-equiv ="scannernavigate" content="[URI]">`	This custom tag allows scanner data to be passed back up to the UCA server. Return Values: Data- The data decoded by the scanner or imaging device. Source- The source device and human readable decoder type of the decoded bar code or symbol. Type- Hex value representing the decoder type. Time- The time at which the decode occurred (hh:mm:ss). Length- The length of the decoded bar code or symbol. Retrieval tags return information by replacing the text '%s' within the content parameter of the meta tag. Each '%s' represents 1 return value, if there are less than the maximum return values allowed for with '%s' place holders, the return values are supplied in sequence for that for the amount supplied. For example: `<meta http-equiv="scannernavigate" content="PriceLookup.htm?label=%s;source=%s;type=%s;time=%s;length=%s" />`
`<meta http-equiv ="device" content="suspend">`	This custom tag causes the device to suspend (sleep) once the business application page is received and parsed.
`<meta http-equiv ="[Symbology name]" content = "[Properties]">`	This custom tag enables/disables symbologies which are not enabled by default. This tag can also set/modify the properties of the symbologies. Parameters: Symbology Name: Any of the symbologies supported by CA 50. Refer to page [scanner symbologies section] Ex: Code128, Codabar, Code11, etc. Properties: properties specific to symbologies Ex: Enable, Disable, minlength, maxlength, etc.

Table 10-1 *HTML Elements Contained in the Head Element (Continued)*

Element	Description
`<meta http-equiv ="reboot" content="warm">`	This custom tag causes the device to reboot once the business application page is received and parsed. The content value can either be "warm" or "cold" for a warm or cold boot, respectively.
`<meta http-equiv ="visibleindicator" content="[indicatortoken]">`	This custom tag causes the LED to blink indefinitely or turn on for the life of the business application page. There are a set of indicator tokens that allow the Web developer to choose the color and blink rate for the LED. These tokens are as follows: [indictortoken] - possible values RedSlowBlink - LED is red and blinks once every second. RedMedBlink - LED is red and blinks once every ½ second. RedFastBlink - LED is red and blinks once every 1/10 second. RedOn - LED is red and stays on. GreenSlowBlink- LED is green and blinks once every second. GreenMedBlink - LED is green and blinks once every ½ second. GreenFastBlink - LED is green and blinks once every 1/10 second. GreenOn - LED is green and stays on. AmberSlowBlink- LED is amber and blinks once every second. AmberMedBlink - LED is amber and blinks once every ½ second. AmberFastBlink - LED is amber and blinks once every 1/10 second. AmberOn - LED is amber and stays on.
`<meta http-equiv ="vibrateindicator" content="[indicatortoken]">`	This custom tag causes the device to vibrate at certain intervals or indefinitely for the life of the business application page. There are set of indicator tokens that allows the Web developer to choose the color, and the blink rate for the LED. These tokens are as follows: [indictortoken] - possible values SlowVibrate - Device vibrates once every second. MedVibrate - Device vibrates once every ½ second. FastVibrate - Device vibrates once every 1/10 second. VibrateOn - Devices vibrates.

HTML Elements Contained in the Body Element

The proprietary HTML elements contained in the body are all custom "input" elements.

Input elements are not rendered even if they are of type text due to the dimensions of the display. The UCA client application behaves as though the custom input tags are visible and "clickable" by a user. Each custom input tag must be within a form element. All input fields that are not of type "submit" or "button" and the input field that was activated are sent to the server using the value of the "action" attribute and the value of the "method" attribute. The form data is sent as GET or POST based on the value of the "method" attribute. POST is used if the "method" attribute is not contained in the form element.

Table 10-2 *HTML Elements Contained in the Body Element*

Element	Description
`<input type=type = "submit" id="ScanData" name="Price" value = "" />`	This input element allows a form to be submitted when the user scans a bar code label. The id attribute's value must be "ScanData" and the type attribute must be "submit" or "button". The value for the "value" attribute is the bar code label scanned when the form is submitted. The name value pair is as follows, "Price=xxxxx" where "xxxxx" is the bar code label scanned.
`<input type="submit" id="LeftSoftKey" name="SoftKey" value="Prev" />`	This input element allow a form to be submitted when the user presses the left soft key on the CA50 device. The id attribute's value must be "LeftSoftKey" and the type attribute must be "submit" or "button". The value for the "value" attribute is the label for the left soft key on the device. Keep in mind that keys defined by the Web application take precedence over the keys defined in the <MenuStates> area of the XML profile. Note: The name attribute's value must be the same as the value for the name attribute of the "RightSoftKey" when both are on a page of the business application.
`<input type="button" id="RightSoftKey" name="SoftKey" value="Next" />`	This input element allows a form to be submitted when the user presses the right soft key on the CA50 device. The id attribute's value must be "RightSoftKey" and the type attribute must be "submit" or "button". The value for the "value" attribute is the label for the right soft key on the device. Keep in mind that keys defined by the Web application take precedence over the keys defined in the <MenuStates> area of the XML profile. Note: The name attribute's value must be the same as the value for the name attribute of the "LeftSoftKey" when both are on a page of the business application. The labels for the soft keys are displayed above the physical buttons on the device. In addition, the soft keys are controlled by the UCA client application if these soft key inputs are not present.

Table 10-2 *HTML Elements Contained in the Body Element (Continued)*

Element	Description

`<input type="submit" id="spinbox" name="USERID" SpinBoxLabel="User ID" SpinBoxLabelPos="beforespinbox" ShowArrows="no" SpinBoxLine="2" SpinBoxMaskDigit ="?" value="??-??-??" LeftBtnLbl ="Next#" RightBtnLbl ="Cancel" />`	The spin box element allows the user to select a series of digits to enter numeric information and submit this information to the server. The left soft key allows the user to move from one numeric digit to the next numeric digit and then wrap when at the last edit field. The **Up** and **Down** keys allow the user to select from zero through nine for each numeric digit which is designated by the "SpinBoxMaskDigit" attribute characters within the "value" attribute. The right soft key cancels the spin box and posts an empty string as the value portion of the name-value pair for this input element. The **Select** or **Enter** key submits the name-value pair for this input element where the value is the contents of the value attribute with each spin box mask digit replaced with a numeric digit that the user keyed in. An explanation and possible values for each attribute are explained below.
	`type`
	This is the standard HTML type attribute for HTML input elements.
	`id`
	This is the standard HTML id attribute for HTML input elements. The contents of the value for this attribute must be "spin box".
	`name`
	This is the standard HTML name attribute for HTML input elements.
	`SpinBoxLabel`
	The contents of this attribute represents the label for the spin box. The content for this attribute is up to the Business Application developer. There is no label if this attribute is not detected for the spin box input element.
	`SpinBoxLabelPos`
	The contents of this attribute represents the spin box label position. The spin box label can be on the line before, the same line, or after the spin box label. Possible values for the "SpinBoxLablePos" attribute can be either "beforespinbox", "sameasspinbox", or "afterspinbox".
	• The *beforespinbox* value inserts the spin box label on the line before the spin box.
	• The *sameasspinbox* value inserts the spin box label on the same line as the spin box which means that there is less space for the spin box label.
	• The *afterspinbox* value inserts the spin box label on the line after the spin box.
	• The default position for the label is the same line as the spin box if the *SpinBoxLabelPos* attribute is not detected in the spin box input element.
	(continued)

Table 10-2 *HTML Elements Contained in the Body Element (Continued)*

Element	Description

(continued)	`ShowArrows`
	The contents of this attribute represents whether or not to put an up and down arrow around the current digit that is being edited. It is only applicable if the spin box is placed on line two. Possible values for "ShowArrows" are "yes" or "no". The default setting for this attribute is "no" if this attribute is not detected.
	`SpinBoxLine`
	The contents of this attribute represents what line number to put on the spin box. The line number can be one, two, or three where line one is the line after the status bar line. The default line for the spin box is line number two if this attribute is not detected in the spin box input element.
	`SpinBoxMaskDigit`
	The contents of this attribute represents what character or characters are mask characters for the value attribute. For example, if this character is a "?" then all characters that are "?" in value attribute is replaced by the character "0" when displayed in the browser window. The user can then edit all these characters by pressing the up or down buttons. The default mask character is the "#" character if this attribute is not detected in the spin box input element.
	`Value`
	The contents of this attribute represents what displays on the CA50 device except the mask characters are replaced by "0" when the spin box is rendered on the browser window. This is explained in more detail in the SpinBoxMaskDigit explanation.
	`LeftBtnLbl`
	The contents of this attribute represents the label of the left soft key. It defaults with the label "next#". The function of the key allows the user to move from each digit by moving one to the right. It wraps back to the first editable digit when the left soft key is pressed and the last editable digit has focus.
	`RightBtnLbl`
	The contents of this attribute represents the label of the right soft key. It defaults with the label "Cancel". The right soft key cancels the spin box and posts an empty string as the value portion of the name-value pair for this input element.

Table 10-2 *HTML Elements Contained in the Body Element (Continued)*

Element	Description

`<input type="Command" name ="PHNPLACECALL" options="1:192.168.0.22:1720:012345" id="RightSoftkey" value ="Expert" >`	This input element allows business application developers to define a soft key for users to press to place a phone call based on the attributes for the "Command" input type.
	`Type ="Command"`
	Command is a key word for the browser. It instructs the browser to execute the command specified in "name" when the specified key is pressed.
	`name ="PHNPLACECALL"`
	PHNPLACECALL is a key word for the browser. It instructs the browser to dial the string found in "options" when the specified key is pressed.
	`options="1:192.168.0.22:1720:012345"`
	Dialing string (DTMF supported format). Options is formatted as follows:
	`PBX (0) or P2P(1):extension:port:DTMF string`
	Where all :'s must be present even if it is PBX and there is no port or if there is no DTMF string Extension-IP address or an actual extension on the PBX Port - for P2P DTMF string.
	`id="RightSoftkey"`
	Specifies the key that initiate the command. Valid values are "RightSoftkey" and "LeftSoftKey".
	`value ="Expert"`
	Text to display over the softkey specified in "id".

Best Practices and Considerations for CA50 Development

There are special considerations to take into account when designing Web pages for the CA50 that might not occur when developing Web pages for desktops or other Motorola mobile devices.

This section addresses CA50 Web page design for data collection and access to the phone and Web based business applications.

Data Collection

Because of the CA50 display size and the number of keys on the device, the following is recommended:

- Collect one piece of user input data per Web page.
- Collect scanner data in a hidden input tag with one scanner input tag per Web page.
- The spin box input element should be the only input element on a form when it is desired to use the spin box input element.
- The select element for a form element can be used and the **Select (S)** key allows the user to submit the current selection. The business application developer can define a soft **Cancel** key so the user can cancel the selected element.

Access to Communication Options and the Web Application

The <MenuStates> area of the XML profile and the Web application together define the key definitions for the user experience. The XML profile definitions typically provide access to the communications options (change walkie-talkie channel, access phone book, hang up phone call … etc.) and links to Web applications. The Web application defines keys for use during a Web based business application.

Design Considerations

Table 10-3 *Design Considerations*

Design Option	Functionality
Whether users should have access to the phone and walkie-talkie settings while they are running Web based business applications.	There are two soft keys available. The business application and communication functionality must share these keys so ensure the business application does not define both soft keys or access to the communication options is unavailable. If a business application screen is a simple *Yes* or *No* question presented to the user, you can define both soft keys for that screen. If the business application is very simple and does not define any keys, spread the communication options across both keys for easy access to certain communication functionality. Be consistent. Through the run of the application, always define the same soft key (right or left) for the application and the other for the communication options. The <MenuStates> to look at are those related to an application running: "AppNoCall" and "AppCallActive". There are two other application related menu states, AppQuietMode and AppCallOnHold, but it is not recommended that the soft key functionality be changed for these states. Pick one key for access to the communications options in the XML profile and leave the other for the business application to use.
Whether users should have access to Web based business applications during a phone call.	If the business application is very simple (does not need to define either of the soft keys) define the <MenuState> key definitions as you wish, maybe optimized for easy access to the phone book. If you want the users to have access to more complex Web based business applications during a phone call, then look at the "DesktopCallActive" and "AppCallActive" <MenuStates>. Again pick one soft key for access to the communications options in the XML profile and leave the other soft key for the business application to use

Example 1 - No Soft Keys Defined by the Web Application

The simplest screen is one in which the Web application defines no keys. An example of this is shown below in *Figure 10-1* with the HTML for that page following. Though the Web application defines no keys, the right soft key is defined as *Cancel*. This key is defined by the UCA client application itself. When the *Cancel* (soft) key is pressed, the UCA client application handles the action and not the browser or the Web application. As part of meta tags data, the code93 symbology is enabled and the maxlength is set to 30.

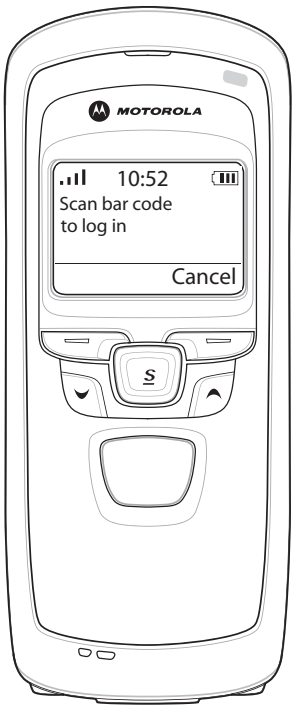


Figure 10-1 *No Soft Keys Defined*

HTML for Web Application Defining No Soft Keys

```
<html xmlns="http://www.w3.org/1999/xhtml" >
    <head>
      <title>UCA User Login</title>
      <META HTTP-Equiv="code93" Content="enable">
      <META HTTP-Equiv="code93-maxlength" Content="30">

    </head>
    <body>
      <form name="form1" method="post" action="GetUser.aspx" id="form1">
        <b>Scan bar code</b><br/>
        <b>to log in</b>
        <input type="text" id="ScanData" size="15" name="userName" value="" runat="server">
    </form>
    </body>
</html>
```

Example 2 - One Soft Key Defined by the Web Application

See *Figure 10-2* and the HTML that follows for a Web page that defines the right soft key as an **Exit** key. This allows the Web application to quit and the UCA client application displays the desktop screen. The left soft key (*Options*) was defined in the XML profile. In this example the client state is "AppNoCall". The user is running a business application and is not on a phone call.

This left soft key allows the user to make a phone call or do other communication functions that are not part of the Web application.

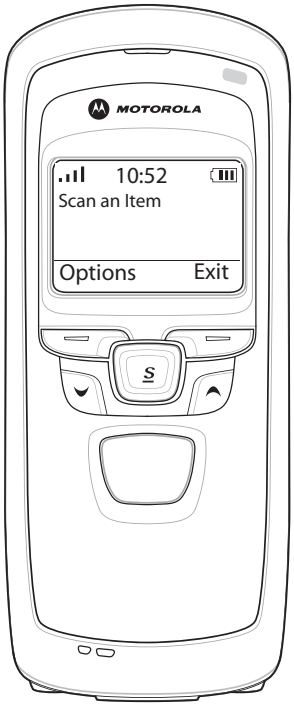


Figure 10-2 *One Soft Key Defined*

HTML for Web Application Defining One Soft Key

```
<html xmlns="http://www.w3.org/1999/xhtml" >
     <head>
       <title>Sample Scanner App</title>
     </head>
     <body>
       <form id="form2" runat="server">
         <input type="text" id="ScanData" size="20" name="item" value="" >
         <input type="submit" name="btn" id="RightSoftKey" value="Exit">
       </form>
    </body>
</html>
```

Example 3 - Both Soft Keys Defined by the Web Application

Figure 10-3 and the HTML that follows shows a Web page where both keys are defined by the Web application. In most cases this would not be a good idea. If the user wanted to change walkie-talkie channels at this moment, he or she could not. In this case, the Web application is expecting a Yes/No response. The *Yes*/*No* response is expected in a short amount of time so it is not a problem that the user cannot get to the UCA client application features at that moment. Using the <font> HTML tag the font can be changed and/or the size increased or decreased. HTML tag <b> can be used to make the text bold.

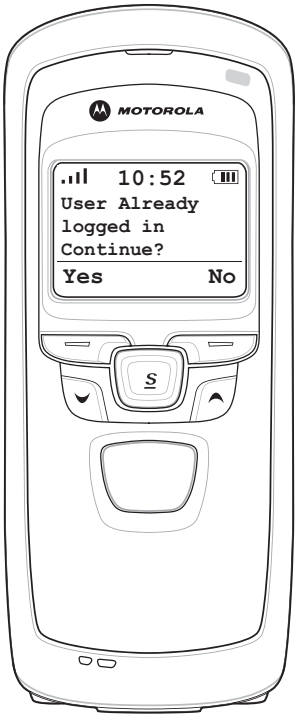


Figure 10-3 *Both Soft Keys Defined*

HTML for Web Application Defining Both Soft Keys

```
<html xmlns="http://www.w3.org/1999/xhtml" >
     <head>
       <title>Dup Login Page</title>
     </head>
     <body>
       <form name="form2" method="post"
action="ConfirmDuplicateLogin.aspx?userName=10017319&sStoreNumber=0002" id="form2">
         <font FACE="courier" size=2>
         <b>User Already</b><br/>
         <b>logged in.</b><br/>
         <b>Continue ?</b>
         </font>
         <input type="submit" name= "btn" id="RightSoftKey" value="No">
         <input type="submit"name= "btn" id="LeftSoftKey" value="Yes">
         <input type="hidden" name= "userName" value="10017319">
         <input type="hidden" name= "sStoreNumber" value="0002">
       </form>
     </body>
</html>
```

Business Applications and their Extensions

During the execution of a Web based business application it may be necessary to interface with the UCA server to query user information or status. This functionality is provided through the Business Application Extensions. This Web service exposes methods that can be called from a Web application.

Business Application Extensions Web Service

In the default installation, the business application extensions are located at:

http://UCAInstallMachine/Motorola/UCA/Client/BAPExtensions.asmx.

Business Application Extensions Methods

UpdateUCAUserSession

This method is used to keep the user's UCA server session alive. The UCA server monitors user activity as it applies to the UCA server. For example, the UCA server recognizes a UCA phone book lookup as user activity. A phone book lookup updates the user's activity in the server and keeps the user's session alive. If the UCA server does not see user activity for a period of time, it logs off the user.

A business application is a completely separate application that may not even reside on the UCA server. If a user only accesses a business application, and never accesses the UCA server (after login), it is possible for the UCA server to log off the user due to lack of activity. The UpdateUCAUserSession method, when called from a business application, updates the user's session in the UCA database.

int UpdateUCAUserSession(string *sStoreNumber*, string *sUser*, ref UCAWebServiceStatus *WSStatus*)

Parameters

sStoreNumber

Store or location number.

sUser

User ID.

WSStatus

UCAWebServiceStatus class describes the status of the call.

Return Values

Returns 1 when successful.

Remarks

Any errors are logged to NT Event log.

GetDeviceUserFromIP

This method is used to lookup a CA50 device user User ID, using the device's IP address as the key.

int GetDeviceUserFromIP(string *sStoreNumber*, string *sIpAddr*, ref string *sUser*,
ref UCAWebServiceStatus *WSStatus*)

Parameters

sStoreNumber

Store or location number.

sIpAddr

IP address of the CA50 device.

sUser

A reference to string that contains the UCA User ID, when successful.

WSStatus

UCAWebServiceStatus class describes the status of the call.

Return Values

Returns 1 and sUser contains the UCA User ID, when successful.

Remarks

Any errors are logged to NT Event log.

(Page intentionally blank.)

Chapter 11 UCA Server Optional APIs

Overview

This chapter outlines how to use the optional APIs that are installed with the UCA server.

Valuable topics in this chapter include:

- UCA server optional APIs overview (*page 11-2*)
- CA50 device user authentication process (*page 11-5*)
- Device user authentication Web service (*page 11-8*)
- Phone extension Web service (*page 11-14*)
- Phone extension Web service methods (*page 11-14*)
- Group management Web service methods (*page 11-18*)
- Role management Web service methods (*page 11-22*)
- Device messaging Web service (*page 11-26*)
- User management Web service (*page 11-29*)
- Console user authentication Web service (*page 11-40*)
- Console user authentication Web service methods (*page 11-40*)
- Customizing the console login process (*page 11-43*)
- UCA server optional APIs samples (*page 11-44*)
- UCA messaging sample (*page 11-44*)
- UCA phone extension sample (*page 11-46*).

UCA Server Optional APIs Overview

The UCA server optional APIs are an alternate administration mechanism enabling a developer to programmatically accomplish the tasks listed below as an alternative to using the UCA server Web Console as the primary data entry interface:

- Login and Authentication of:
  - CA50 users (see *page 11-5*)
  - Web Console Users (see *page 11-40*)
- Server Management and Maintenance of Customer Data
  - Phone extensions (see *page 11-14*)
  - Groups (see *page 11-18*)
  - Roles (see *page 11-22*)
  - Device Users (see *page 11-29*)
  - Console Users (see *page 11-36*)
- Sending Text Messages to the CA50 (see *page 11-26*)

The Application Programming Interfaces (APIs) in this chapter can be utilized to extend the standard UCA server functionality provided from Motorola. These APIs enable integration of the UCA server into existing store processes. The use of these APIs is optional.

These optional APIs can be used by customers or third party programmers to customize the login process, assist in the creation of external programs that might be used to initialize or maintain the UCA server data and send text messages. The APIs are implemented using Web services.

Location of Optional APIs and Web Messaging Service

The default authentication Web services are located in the /Motorola/UCA/Client/Authentication and the /Motorola/UCA/ServerManagement/Authentication directories under IIS. During the UCA login processes, these Web services are called to authenticate the user's credentials against the user data found in the UCA database. The system can be configured to call a custom authentication Web service that authenticates against an entirely different database, if needed.

The server management and maintenance Web services are located in the /Motorola/UCA/ServerManagement/ directory under IIS. These Web services can be used to populate the UCA server database with key information, such as device users, phone extensions, etc. External programs can be developed that utilize the functionality found in these Web services as an alternative or in addition to using the UCA server Web Console in some cases. For example, if a customer finds over time that they need more CA50s in a given department (e.g., Lumber) these new devices most likely need phone extensions. If the number of CA50s in Lumber exceeds the number of phone extensions that were allocated to the Lumber group in the UCA server, more phone extensions need to be assigned to Lumber. The assignment of these new phone extensions to the Lumber group in the UCA server database can be accomplished by using the Web Console phone extension GUI functionality, or by creating an external program that uses the AddExtension() function found in the Phone Extension Management Web service.

The messaging Web services are located in the /Motorola/UCA/ServerManagement directory under IIS. This Web service can be used by external programs to send asynchronous messages to logged on CA50 users.

Alternate Mechanism for Login and Authentication

Device user authentication is based on the user information on the UCA server. Alternatively, customers with an existing database that contains user (staff) information can use their existing user authentication mechanism. Note that the customer's authentication process is used only for authentication purposes; other user information must be stored in the UCA database, such as the user's group and role.

In this case, the customer can use a Web service with the same signature as that of the UCA Authentication web service, and include this in the web.config. See *Customizing the Process Flow on page 11-6* for information on updating web config.

c:/Inetpub/wwwroot/Motorola/UCA/Client\web.config

```
<applicationSettings>
<UCACommon.Properties.Settings>
<setting name="UCACommon_localhost_Authentication" serializeAs="String">
<value>http://localhost/Motorola/UCA/Client/Authentication/Authentication.asmx</value>
</setting>
</UCACommon.Properties.Settings>
</applicationSettings>
```

Similarly, if a customer's database contains the console users' (typically administrators') information, the customer must provide an alternative web service similar to the one UCA uses for console user authentication in order to use this for console user authentication. To do this, the customer must change the URL in the web.config to point to his own Web service. See *Customizing the Console Login Process on page 11-43* for more information on updating Web config.

C:\Inetpub\wwwroot\Motorola\UCA\UCAConsole\web.config

```
<applicationSettings>
<UCACommon.Properties.Settings>
<setting name=" UCACommon_ConsoleAuthService_ManagementAuthentication" serializeAs="String">
<value>http://localhost/Motorola/UCA/ServerManagement/Authentication/
ManagementAuthentication.asmx </value>
</setting>
</UCACommon.Properties.Settings>
</applicationSettings>
```

Alternative Mechanism for Server Management and Maintenance of Customer Data

Programmers can use the UCA server optional APIs to port existing customer database(s) into the UCA server's database rather than using the Web Console to enter user data such as phone book extensions, user information, groups, and roles. This simplifies and speeds the entry and maintenance of data in the UCA server by leveraging the same API used by the Web Console to interface to the UCA server's databases.

Mechanism for Sending Text Messages to the Device

You must use a Web service to send text messages from the UCA server to the CA50 (Asynchronous Messaging Mechanism). See *CA50 Features on page 1-2* for an overview of the CA50 text messaging capabilities.

Phone Extension

To enter phone extension data into the UCA server, go to:

http://<*UCAInstallMachine*>/Motorola/UCA/Samples/UCAPhoneExtensions/default.aspx

where *UCAInstallMachine* is the UCA server IP address.

Messaging

To enter messaging information into the UCA server, go to:

http://<*UCAInstallMachine*>/Motorola/UCA/Samples/UCAMessaging/default2.aspx

where *UCAInstallMachine* is the UCA server IP address.

Source: C\Program Files\Motorola\UCA Server\DevelopmentToolkit\PhoneExtensions.

CA50 Device User Authentication Process

The UCA server uses a combination of .NET Web services and .NET aspx pages for the CA50 device user Authentication Process.

Process Flow

Upon startup, the CA50 client calls the DeviceRegistration method in the DeviceManagement Web Service to register with the UCA system. The DeviceManagement Web Service calls the AuthenticateMAC method in the Authentication Web Service to potentially authenticate the device MAC address and determine the next step for the UCA Authentication Process. The MAC level authentication is only attempted if the feature is enabled in the system settings. In most cases this is not used because of the operational work involved in tracking the MAC address of each device in an enterprise.

MAC authentication is enabled by a setting in the UCA database (DB). If enabled, the AuthenticateMAC method attempts to match the MAC of the device as an external user ID to the entries in the UCA B_User table. To convey the status of the authentication attempt and the next step in the overall UCA Authentication Process, the AuthenticateMAC method returns a class object (AuthenticateResponse) that contains the member variables:

authenticated (boolean)
continueOn (boolean)
token (string) - Used to get a URL from the UCA DB

The DeviceRegistration method evaluates the return values from AuthenticateMAC in the following ways.

Example 1 - Authentication Fails, Process Ends

authenticated = false
continueOn = false
token (string) = ""

Return values in this example indicate that the device is not authenticated and the UCA Authentication Process should not continue. The device failed authentication at the MAC level and the process ends.

Example 2 - Authentication Successful, Process Ends

authenticated = true
continueOn = false
token (string) = ""

Return values in this example indicate that the device is authenticated and the overall UCA Authentication Process is complete. The device passed MAC authentication. This is sufficient. and the process ends.

Example 3 - Authentication Requires Additional Information, Process Continues

authenticated = false
continueOn = true
token (string) = "GetUserID"

Return values in this example indicate that the UCA Authentication Process is not over. The token string “GetUserID” represents the next step in the UCA Authentication Process. This token is used to look up a URL in the UCA DB System Settings. The token string appends to "System.Login.Token." and the resulting string is looked up in the S_SystemVariables table for a reference to the B_SystemSettings table (svID). The svID is looked up in the B_SystemSettings to get the URL.

The DeviceRegistration Web Service returns the URL to the device to which it browses. In the default DB configuration, the token maps to "GetUser.aspx." The code behind GetUser.aspx calls the AuthenticateUser method, also in the Authentication Web Service. The return values from AuthenticateUser are utilized in the same manner as the response from AuthenticateMAC described above. The return values indicate whether or not the process is complete and if not report the next step. This process can continue to collect a password, a group (department), etc. from the user during login. This process is configurable.

When the device or user is finally validated, a user profile is generated and embedded in a Web page or returned to the device in response to the DeviceRegistration Web Service call. See *Figure 11-1 on page 11-7* for a diagram of the UCA Authentication Process.

Customizing the Process Flow

The default UCA Authentication Web Service used in the UCA Authentication Process can be replaced by an Authentication Web Service written by a customer or integrator.

This can be accomplished as follows:

1. Create an Authentication Web Service that matches the signature, methods and return values of the UCA Authentication Web Service.
2. Change the Web.config Web reference to point to the new Authentication Web Service.
3. Modify the system settings login tokens to point to the new aspx pages.

Motorola provides an example Authentication Web Service with method stubs that can be used as a starting point for a customer or integrator who wishes create their own.

To change the Web.config to use a different Authentication Web Service, find the following section in the Web.config file:

```
<applicationSettings>
    <UCACommon.Properties.Settings>
      <setting name="UCACommon_localhost_Authentication" serializeAs="String">
<value>http://localhost/Motorola/UCA/Client/Authentication/Authentication.asmx</value>
      </setting>
    </UCACommon.Properties.Settings>
  </applicationSettings>
```

In the default installation, Web.config is located in: c:/Inetpub/wwwroot/Motorola/UCA/Client.

Change the <value> tag above to point to the new Authentication Web Service and the UCA server calls the new Authentication Web Service during the UCA Authentication Process. This setting may be encrypted for security purposes. See *UCA Server Security on page 6-54* for more information.

UCA Authentication Process Diagram

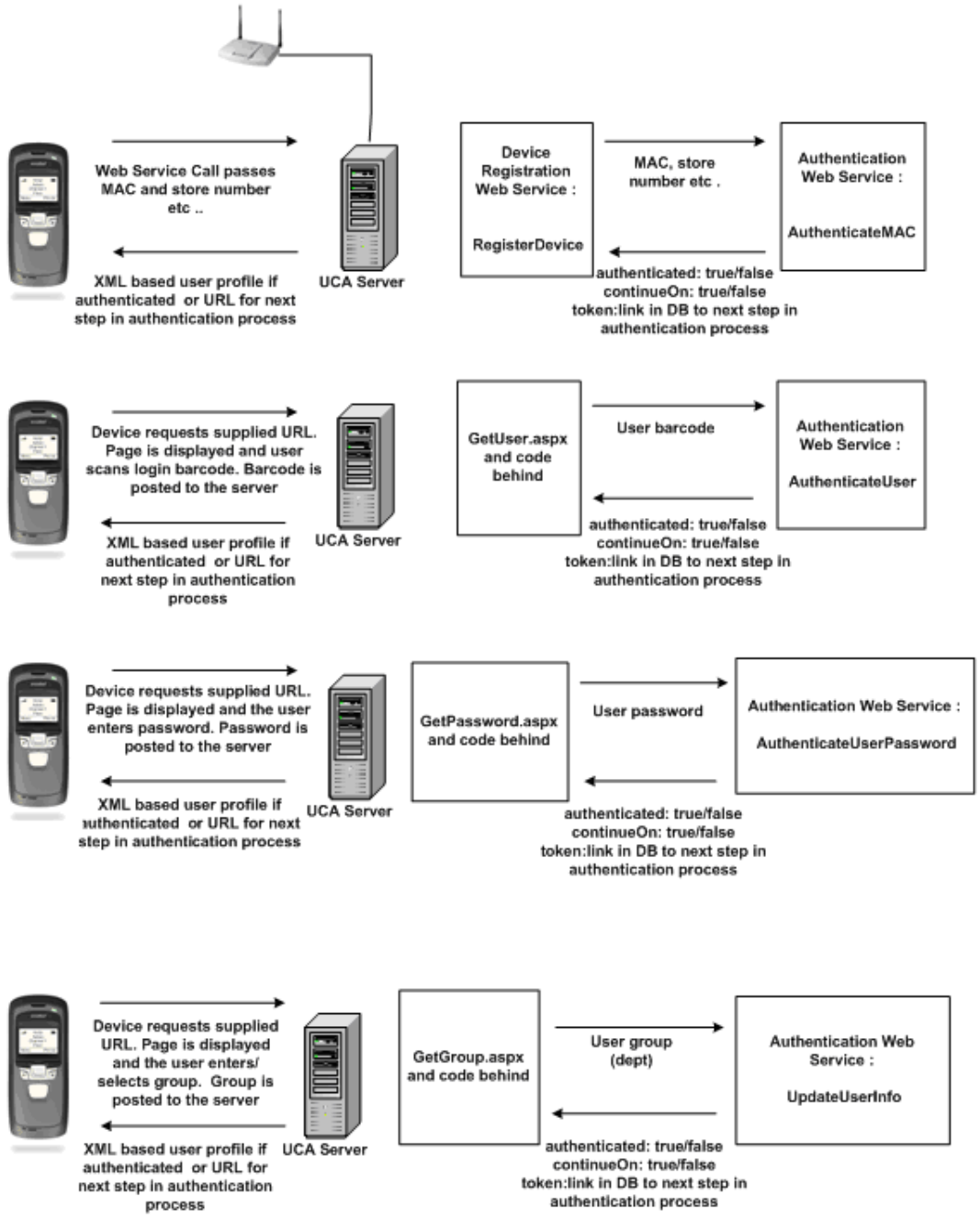

Web Service Call passes MAC and store number etc ..
XML based user profile if authenticated or URL for next step in authentication process
UCA Server
Device Registration Web Service :
RegisterDevice
MAC, store number etc .
Authentication Web Service :
AuthenticateMAC
authenticated: true/false
continueOn: true/false
token:link in DB to next step in authentication process
Device requests supplied URL. Page is displayed and user scans login barcode. Barcode is posted to the server
XML based user profile if authenticated or URL for next step in authentication process
UCA Server
GetUser.aspx and code behind
User barcode
Authentication Web Service :
AuthenticateUser
authenticated: true/false
continueOn: true/false
token:link in DB to next step in authentication process
Device requests supplied URL. Page is displayed and the user enters password. Password is posted to the server
XML based user profile if authenticated or URL for next step in authentication process
UCA Server
GetPassword.aspx and code behind
User password
Authentication Web Service :
AuthenticateUserPassword
authenticated: true/false
continueOn: true/false
token:link in DB to next step in authentication process
Device requests supplied URL. Page is displayed and the user enters/ selects group. Group is posted to the server
XML based user profile if authenticated or URL for next step in authentication process
UCA Server
GetGroup.aspx and code behind
User group (dept)
Authentication Web Service :
UpdateUserInfo
authenticated: true/false
continueOn: true/false
token:link in DB to next step in authentication process

Figure 11-1 *Client/Server UCA Authentication Process Flow*

Device User Authentication Web Service

In the default installation, the Device User Authentication Web Service is located at:

http://UCAInstallMachine/Motorola/UCA/Client/Authentication/Authentication.asmx.

This Web service is used in the authentication process of a CA50 device user.

Authentication Web Service Methods

GetUserData

This method retrieves user information from the UCA system. The user's information is updated in the UserInfo class passed to the method.

int GetUserData(string *sStoreNumber,* string *sUser*, string *ipAddr*, ref *UserInfo* userInfo, ref UCAWebServiceStatus *WSStatus)*

Parameters

sStoreNumber

User's store number.

sUser

User's login ID.

ipAddr

IP address of source machine (optional - used for logging).

Pass "" when address logging is not required.

UserInfo

A reference to a UserInfo class that describes the user (described above).

WSStatus

UCAWebServiceStatus class describes the status of the call.

Return Values

Returns a value of 1 when successful. When successful, UserInfo is populated with the specified user's information.

Remarks

Any errors are logged to the NT Event log.

Related Classes

```
public class UserInfo
{
  public UserInfo(){}
  public string ExtUsrID;
  public string firstName;
  public string lastName;
  public string userName;
  public string RoleToken;
  public string locExtID;
  public string userStatus;
  public string userPhExtn;
  //Groups are listed by Rank
  public UserGroupInfo[] aUserGroups;
}
public class UserGroupInfo
{
  public UserGroupInfo(){}
  public string GroupToken;
  public int Rank;
  public bool IsSelectable;
  public string GroupName;
}
```

UserInfo Class Members

Table 11-1 *UserInfo Class Members*

Class Member	Description
ExtUsrID	User login ID.
firstName	User first name.
lastName	User last name.
userName	User display name. Format = last name, first initial.
RoleToken	Token representing user role (e.g., Manager, Sales, FrontEnd, etc.). This data is found in the B_Roles table.
locExtID	Location ID (store number).
userStatus	User status (e.g., Active, Suspended, Disabled). This data is found in the S_UserStatusType table.
userPhExtn	User phone extension (if user is logged into server). If not available the value is NA

UserGroupInfo Class Members

Table 11-2 *UserGroupInfo Class Members*

Class Member	Description
GroupToken	Token for the department. This data is found in the B_Groups table.
Rank	Group rank. Used for presenting multiple groups to a user. A rank of 1 displays first in a list.
IsSelectable	Indicates whether or not a group can be selected in a list presented to a user.
GroupName	Display name of a group.

The WSStatus variable should be examined in cases where the return value from the Web Service indicates a failure.

```
public class UCAWebServiceStatus
{
   public UCAWebServiceStatus (){}
     public int returnStatus;
     public string sMessage;
}
```

Table 11-3 *UCAWebService Class Members*

Class Member	Description
returnStatus	Error code.
sMessage	Message describing the error.

```
    }
UCAWebService Class Members
returnStatus
    Error Code
sMessage
    Message describing the error
```

AuthenticateMAC

This method is used to authenticate a device. If authMac required is enabled, the MAC is looked up as a user in the B_USER table.

int AuthenticateMAC(string sStoreNumber, string sMAC, string ipAddr ref AuthenticateResponse AuthResp, ref UCAWebServiceStatus WSStatus)

Parameters

sStoreNumber

Device store number.

sMAC

Device MAC address.

ipAddr

IP address of source machine (optional - used for logging).

Pass "" when address logging is not required.

AuthResp

Result of authentication attempt. (See *CA50 Device User Authentication Process on page 11-5*.)

WSStatus

UCAWebServiceStatus class describes the status of the call.

Return Values

Returns 1 when successful. If successful, AuthResp is populated with the result of the authentication attempt. If return value is 0, check WSStatus for error code and description.

Remarks

Any errors are logged to the NT Event log.

Related Classes

```
public class AuthenticateResponse
{
  public AuthenticateResponse (){}
  public bool     authenticated;
  public bool     continueOn;
  public string   token;
}
```

AuthenticateResponse Class Members

Table 11-4 *AuthenticateResponse Class Members*

Class Member	Description
authenticated	Is the user authenticated?
continueOn	Are there more steps to the UCA Authentication Process?
token	DB token representing the next step in the UCA Authentication Process, if there is another step.

AuthenticateUser

This function attempts to authenticate a CA50 device user.

int AuthenticateUser(string *sStoreNumber*, string *sUser*, string *ipAddr*, ref AuthenticateResponse *AuthResp*, ref UCAWebServiceStatus *WSStatus*)

Parameters

sStoreNumber

User's store number.

sUser

User's login ID.

ipAddr

IP address of source machine (optional - used for logging).

Pass "" when address logging is not required.

AuthResp

Result of authentication attempt. (See *CA50 Device User Authentication Process on page 11-5*.)

WSStatus

UCAWebServiceStatus class describes the status of the call.

Return Values

Returns a value of 1 when successful. If successful, AuthResp is populated with the result of the authentication attempt. If the return value is 0, check WSStatus for the error code and description.

Remarks

Any errors are logged to the NT Event log.

AuthenticateUserPassword

This function attempts to authenticate a CA50 device user and password.

int AuthenticateUserPassword(string *sStoreNumber*, string *sUser*, string *sPassword*, string *ipAddr*, ref AuthenticateResponse *AuthResp*, ref UCAWebServiceStatus *WSStatus*)

Parameters

sStoreNumber

User store number.

sUser

User login ID.

sPassword

User password.

ipAddr

IP address of source machine (optional - used for logging).

Pass "" when address logging is not required.

AuthResp

Result of authentication attempt. (See *CA50 Device User Authentication Process on page 11-5*.)

WSStatus

UCAWebServiceStatus class describes the status of the call

Return Values

Returns a value of 1 when successful. If successful, AuthResp is populated with the result of the authentication attempt. If the return value is 0, check WSStatus for the error code and description.

Remarks

Any errors are logged to the NT Event log.

Phone Extension Web Service

In the default installation the Phone Extension Management Web Service is located at:

http://UCAInstallMachine/Motorola/UCA/ServerManagement/PhoneExtensionManagement.asmx.

This Web service can be used by external programs to modify the phone extension data in the UCA database.

Phone Extension Web Service Methods

This function adds a phone extension that can be used by a CA50 device. the extensions are added to the B_PhoneExt table.

int AddExtension(PhoneExtInfo *phExtInfo*, string *ipAddr*, string *sUser*, ref UCAWebServiceStatus *WSStatus*)

Parameters

phExtInfo

Phone extension info class that describes the phone extension

ipAddr

IP address of source machine (optional - used for logging).

Pass "" if do not wish to have the address logged.

sUser

UserID of user making WS call (optional - used for logging).

Pass "" when user logging is not required.

WSStatus

UCAWebServiceStatus class describes the status of the call

Return Values

Returns 1 when successful.

Remarks

Any errors are logged to the NT Event log.

Related Classes

```
public class PhoneExtInfo
    {
      public PhoneExtInfo(){}
      string phnExtPhoneNumber,
      string phnDesc,
      string phnOwnerID,
      string phnRangeToken,
      string locExtID,
      bool phnIsAlwaysShown,
      string phnExtensionPassword,
      string phnExtType
      public string phnMAC,
      public bool phnInvalid,
      public string
      phnInvalidReason
    }
```

phoneExtInfo Class Members

Table 11-5 *phoneExtInfo Class Members*

Class Member	Description
phnExtPhoneNumber	Phone extension number.
phnDesc	Textual description of the phone extension.
phnOwnerID	UserID or MAC for static UCA assignment.
phnRangeToken	Token for department. Found in the B_Groups table.
locExtID	Store number.
phnIsAlwaysShown	Always show in phone book regardless of login status.
phnExtensionPassword	PB.X registration password
phnExtType	Extension type: UCA Dynamic, UCA Static, Non-UCA. Found in the S_PhoneExtType table.
phnMAC	The MAC address used for SCCP.
phnInvalid	The extension is either a valid SCCP extension or it is not valid.
phnInvalidReason	The reason for why the extension is not valid.

RemoveExtension

This method removes a phone extension from the UCA system. The extensions are removed from the B_PhoneExt table.

int RemoveExtension(string *locExtID*, string *sPhnExt*, string *ipAddr*, string *sUser*, ref UCAWebServiceStatus *WSStatus*)

Parameters

locExtID

Store number.

sPhnExt

Phone extension number.

ipAddr

IP address of source machine (optional - used for logging).

Pass "" when address logging is not required.

sUser

UserID of user making WS call (optional - used for logging).

Pass "" when user logging is not required.

WSStatus

UCAWebServiceStatus class describes the status of the call.

Return Values

Returns 1 when successful.

Remarks

Any errors are logged to the NT Event log.

UpdateExtension

This method updates phone extension information in the UCA system. The extension information is updated in the the B_PhoneExt table in the UCA database.

int UpdateExtension(PhoneExtInfo *phExtInfo*, string *ipAddr*, string *sUser*,
ref UCAWebServiceStatus *WSStatus*)

Parameters

phExtInfo

Phone extension info class that describes the phone extension.

ipAddr

IP address of source machine (optional - used for logging).

Pass "" when address logging is not required.

sUser

UserID of user making WS call (optional - used for logging).

Pass "" when user logging is not required.

WSStatus

UCAWebServiceStatus class describes the status of the call.

Return Values

Returns 1 when successful.

Remarks

Any errors are logged to the NT Event log.

GetExtensionInfo

This method retrieves phone extension information from the UCA system. The extension information is populated into the PhoneExtInfo class passed to the method.

public int GetExtensionInfo(string *locExtID*, string *sPhnExt*, string *ipAddr*, string *sUser*, ref PhoneExtInfo *phExtInfo*, ref UCAWebServiceStatus *WSStatus*)

Parameters

locExtID

storeNumber.

sPhnExt

Phone extension.

ipAddr

IP address of source machine (optional - used for logging).

Pass "" when address logging is not required.

sUser

UserID of user making WS call (optional - used for logging).

Pass "" when user logging is not required.

PhoneExtInfo

Reference to a PhoneExtInfo class (described above)

WSStatus

UCAWebServiceStatus class describes the status of the call

Return Values

Returns 1 when successful. When successful, phExtInfo is populated with the specified extension information.

Remarks

Any errors are logged to the NT Event log.

Group Management Web Service Methods

In the default installation the Group Management Web Service is located at:

http://UCAInstallMachine/Motorola/UCA/ServerManagement/GroupManagement.asmx.

This Web service can be used by external programs to modify the UCA group data in the UCA database.

AddGroup

This method adds a Group (Dept.) into the UCA system. The groups are added into the B_Groups table.

int AddGroup(string *sStoreNumber*, GroupInfo *groupInfo*, string *ipAddr*, string *sUser*, ref UCAWebServiceStatus *WSStatus*)

Parameters

sStoreNumber

Store Number

groupInfo

Group Information info class that describes the organizational group (department).

ipAddr

IP address of source machine (optional - used for logging).

Pass "" when address logging is not required.

sUser

UserID of user making WS call (optional - used for logging).

Pass "" when user logging is not required.

WSStatus

UCAWebServiceStatus class describes the status of the call.

Return Values

Returns 1 when successful.

Remarks

Any errors are logged to the NT Event log.

Related Classes

```
public class GroupInfo
{
  public GrouptInfo(){}
  string GroupToken,
  string GroupName,
  string GroupExtID,
  bool IsSelectable,
}
```

GroupInfo Class Members

Table 11-6 *GroupInfo Class Members*

Class Member	Description
GroupToken	Group ID. Must be unique.
GroupName	Group name for display purposes.
GroupExtID	Group number
IsSelectable	Is this group selectable by the user at login?

RemoveGroup

This method removes a group from the UCA system. The groups are removed from the B_Groups table.

int RemoveGroup(string *sStoreNumber*, string *sGroupToken*, string *ipAddr*, string *sUser*, ref UCAWebServiceStatus *WSStatus*)

Parameters

sStoreNumber

Store number.

sGroupToken

Group ID. Identifier for the Group.

ipAddr

IP address of source machine (optional - used for logging).

Pass "" when address logging is not required.

sUser

UserID of user making WS call (optional - used for logging).

Pass "" when user logging is not required.

WSStatus

UCAWebServiceStatus class describes the status of the call.

Return Values

Returns 1 when successful.

Remarks

Any errors are logged to the NT Event log.

UpdateGroup

This method updates a Group's information in the UCA system. The Group's information is updated in the B_Groups table in the UCA database.

int UpdateGroup(string *sStoreNumber*, GroupInfo *groupInfo*, string *ipAddr*, string *sUser*, ref UCAWebServiceStatus *WSStatus*)

Parameters

sStoreNumber

Store Number

groupInfo

Group Information info class that describes the organizational group (department).

ipAddr

IP address of source machine (optional - used for logging).

Pass "" when address logging is not required.

sUser

UserID of user making WS call (optional - used for logging).

Pass "" when user logging is not required.

WSStatus

UCAWebServiceStatus class describes the status of the call.

Return Values

Returns 1 when successful.

Remarks

Any errors are logged to the NT Event log.

GetGroupInfo

This method retrieves group information from the UCA system.

public int GetGroupInfo(string *sStoreNumber*, string *sGroupToken*, string *ipAddr*, string *sUser*, ref PhoneExtInfo *groupInfo*, ref UCAWebServiceStatus *WSStatus*)

Parameters

sStoreNumber

Store number.

sGroupToken

Group ID. Identifier for the Group.

ipAddr

IP address of source machine (optional - used for logging).

Pass "" when address logging is not required.

sUser

UserID of user making WS call (optional - used for logging).

Pass "" when user logging is not required.

groupInfo

Reference to a groupInfo class (see *Table 11-6 on page 11-19*).

WSStatus

UCAWebServiceStatus class describes the status of the call.

Return Values

Returns 1 when successful. When successful, groupInfo is populated with the specified group information.

Remarks

Any errors are logged to the NT Event log.

Role Management Web Service Methods

In the default installation the Role Management Web Service is located at:

http://UCAInstallMachine/Motorola/UCA/ServerManagement/RoleManagement.asmx.

This Web service can be used by external programs to modify the UCA role data in the UCA database.

AddRole

This method adds a role into UCA system. The roles are added into the B_Roles table.

int AddRole(string *sStoreNumber*, RoleInfo *roleInfo*, string *ipAddr*, string *sUser*, ref UCAWebServiceStatus *WSStatus*)

Parameters

sStoreNumber

Store Number.

roleInfo

Role Information info class that describes the organizational role.

ipAddr

IP address of source machine (optional - used for logging).

Pass "" when address logging is not required.

sUser

UserID of user making WS call (optional - used for logging).

Pass "" when user logging is not required.

WSStatus

UCAWebServiceStatus class describes the status of the call.

Return Values

Returns 1 when successful.

Remarks

Any errors are logged to the NT Event log.

Related Classes

```
public class RoleInfo
{
  public RoleInfo(){}
  string RoleToken,
  string RoleName,
  string RoleExtID,
}
```

RoleInfo Class Members

Table 11-7 *RoleInfo Class Members*

Class Member	Description
RoleToken	Role ID. Must be unique.
RoleName	Role name for display purpos.es
RoleExtID	Role number.

RemoveRole

This method removes a role from the UCA system. The roles are removed from the B_Roles table.

int RemoveRole(string *sStoreNumber*, string *sRoleToken*, string *ipAddr*, string *sUser*, ref UCAWebServiceStatus *WSStatus*)

Parameters

sStoreNumber

Store number.

sRoleToken

Role ID. Identifier for the Role.

ipAddr

IP address of source machine (optional - used for logging).

Pass "" when address logging is not required.

sUser

UserID of user making WS call (optional - used for logging).

Pass "" when user logging is not required.

WSStatus

UCAWebServiceStatus class describes the status of the call.

Return Values

Returns 1 when successful.

Remarks

Any errors are logged to the NT Event log.

UpdateRole

This method updates a role's information in the UCA system. The role's information is updated in the B_Roles table in the UCA database.

int UpdateRole(string *sStoreNumber*, RoleInfo *roleInfo*, string *ipAddr*, string *sUser*, ref UCAWebServiceStatus *WSStatus*)

Parameters

sStoreNumber

Store Number.

roleInfo

Role Information info class that describes the organizational role.

ipAddr

IP address of source machine (optional - used for logging).

Pass "" when address logging is not required.

sUser

UserID of user making WS call (optional - used for logging).

Pass "" when user logging is not required.

WSStatus

UCAWebServiceStatus class describes the status of the call.

Return Values

Returns 1 when successful.

Remarks

Any errors are logged to the NT Event log.

GetRoleInfo

This method retrieves role information from the UCA system.

public int GetRoleInfo(string *sStoreNumber*, string *sRoleToken*, string *ipAddr*, string *sUser*, ref RoleInfo *roleInfo*, ref UCAWebServiceStatus *WSStatus*)

Parameters

sStoreNumber

Store number.

sRoleToken

Role ID. Identifier for the Role.

ipAddr

IP address of source machine (optional - used for logging).

Pass "" when address logging is not required.

sUser

UserID of user making WS call (optional - used for logging).

Pass "" when user logging is not required.

roleInfo

Reference to a roleInfo class (see *Table 11-7 on page 11-23*).

WSStatus

UCAWebServiceStatus class describes the status of the call

Return Values

Returns 1 when successful. When successful, roleInfo is populated with the specified role information.

Remarks

Any errors are logged to the NT Event log.

Device Messaging Web Service

In the default installation the Device Messaging Web Service is located at:

http://UCAInstallMachine/Motorola/UCA/ServerManagement/DeviceMessaging.asmx.

This Web service can be used by external programs to send an asynchronous message to a logged in CA50 device user.

Device Messaging Web Service Methods

SendNotification

This method sends an asynchronous message to a CA50 device.

Int SendNotification(string *sStoreNumber*, NotifDest *destInfo*, DynamicNotifParams *notifInfo*, string *ipAddr*, string *sUser*, ref UCAWebServiceStatus *WSStatus*)

Parameters

sStoreNumber

Store Number.

destInfo

Destination information class that describes where the message is sent.

notifInfo

Notification information provides details about the notification.

ipAddr

IP address of source machine (optional - used for logging). Pass "" when address logging is not required.

sUser

UserID of user making WS call (optional - used for logging). Pass "" when user logging is not required.

WSStatus

UCAWebServiceStatus class describes the status of the call.

Related Classes

```
public class NotifDest
{
    public NotifDest(){}
    string sDestID,
    string eDestType,
    MRSType eMRSType;
}
```

NotifDest Class Members

Table 11-8 *NotifDest Class Members*

Class Member	Description
sDestID	Destination ID: UserID, IP address and group. This works with DestinationType. If eDestType is User, this param should be set to the UserID of the person to which the message is sent. If eDestType is group, this parameter is set to the (where the message is sent) group name.

eDestType	Destination Type. For example: User, IP address, etc. Use emun DestTypes as follows:
	`public enum DestTypes`
	`{`
	`Role`
	`Group`
	`User`
	`DeviceID,`
	`DeviceIPAddr,`
	`WTChannel,`
	`All`
	`}`
	`public class DynamicNotifParams`
	`{`
	`public DynamicNotifParams(){}`
	`string Text,`
	`string tone,`
	`string VisibleIndicator,`
	`string VibrateIndicator,`
	`string AdditionalAttribs,`
	`string rskey,`
	`string lskey`
	`{`

eMRSType	Type of Media Relay Service to use. IPBLUE_MRS for IP Blue Media Relay Service, CA50_MRS for internal purpose, OTHER_MRS is currently not supported. Use the CA50_MRS when the message length is longer than 140 characters.
	`public enum MRSType {`
	`CA50_MRS,`
	`IPBLUE_MRS,`
	`OTHER_MRS,`
	`}`

DynamicNotifParams Class Members

Table 11-9 *DynamicNotifParams Class Members*

Class Member	Description
Text	Text to be displayed on the device.
tone	Path on the device for the tone to be played.
VisibleIndicator	Led color and blink rate. Visible Indicator Values: • REDSLOWBLINK • REDMEDBLINK • REDFASTBLINK • REDON • GREENSLOWBLINK • GREENMEDBLINK • GREENFASTBLINK • GREENON
VibrateIndicator	Vibrate action. Vibrate Indicator Values: • SLOWVIBRATE • MEDVIBRATE • FASTVIBRATE • VIBRATEON
VisibleCycleCount	The number of times the device should vibrate on receipt of notification message. If it is passed as null, the value is picked form the XML profile (NotifMsgVibrateCycleCount).
VibrateCycleCount	The number of times the device should show visible indication on receipt of notification message. If it is passed as null, the value is picked form the XML profile (NotifMsgVisibleCycleCount).
lskey	Action associated with left soft key. Ex: <add key="LSKey" value="Ack:S:S:http://192.168.7.138/HRA/A.aspx?requestInfo=1100:23" /> For more information see *Menu Elements on page 7-15*.
rskey	Action associated with right soft key. Ex: <add key="RSKey" value="Dismiss:S:S:http://192.168.7.138/HRA/N.aspx?requestInfo=1100:23" /> For more information see *Menu Elements on page 7-15*.

User Management Web Service

In the default installation the UserManagement Web Service is located at:

http://UCAInstallMachine/Motorola/UCA/ServerManagement/UserManagement.asmx.

This Web service can be used by external programs to modify UCA user records in the UCA database.

User Management Web Service Methods

AddDeviceUser

This method adds a device user to the UCA system. The user is added into the B_User table in the UCA database.

int AddDeviceUser(UserInfo *userInfo*, string *sPIN*, string *ipAddr*, ref UCAWebServiceStatus *WSStatus*)

Parameters

userInfo

UserInfo class that describes the user.

sPIN

User PIN.

ipAddr

IP address of source machine (optional - used for logging).
Pass "" if the address should not be logged.

WSStatus

UCAWebServiceStatus class describes the status of the call.

Return Values

Returns a value of 1 when successful.

Remarks

Any errors are logged to the NT Event log.

Related Classes

```
public class UserInfo
{
  public UserInfo(){}
  public string ExtUsrID;
  public string firstName;
  public string lastName;
  public string userName;
  public string RoleToken;
  public string locExtID;
  public string userStatus;
  public string userPhExtn;
  //Groups are listed by Rank
  public UserGroupInfo[] aUserGroups;
}
public class UserGroupInfo
{
  public UserGroupInfo(){}
  public string GroupToken;
  public int Rank;
  public bool IsSelectable;
  public string GroupName;
}
```

UserInfo Class Members

Table 11-10 *UserInfo Class Members*

Class Member	Description
ExtUsrID	User login ID.
firstName	User first name.
lastName	User last name.
userName	User display name. Format = last name, first initial.
RoleToken	Token representing user role (e.g., Manager, Sales, FrontEnd, etc.). This data is found in the B_Roles table.
locExtID	Location ID (store number).
userStatus	User status (e.g., Active, Suspended, Disabled). This data is found in the S_UserStatusType table.
userPhExtn	User phone extension (if user is logged into server). If not available the value is NA

UserGroupInfo Class Members

Table 11-11 *UserGroupInfo Class Members*

Class Member	Description
GroupToken	Token for the department. This data is found in the B_Groups table.
Rank	Group rank. Used for presenting multiple groups to a user. A rank of 1 displays first in a list.
IsSelectable	Indicates whether or not a group can be selected in a list presented to a user.
GroupName	Display name of a group.

The WSStatus variable should be examined in cases where the return value from the Web Service indicates a failure.

```
public class UCAWebServiceStatus
{
   public UCAWebServiceStatus (){}
     public int returnStatus;
     public string sMessage;
}
```

Table 11-12 *UCAWebService Class Members*

Class Member	Description
returnStatus	Error code.
sMessage	Message describing the error.

```
   }
UCAWebService Class Members
returnStatus
   Error Code
sMessage
   Message describing the error
```

RemoveDeviceUser

This method removes a device user from the UCA system. The user is removed from the B_USER table in the UCA database.

int RemoveDeviceUser(string *sStoreNumber*, string *sUser*, string *ipAddr,* ref UCAWebServiceStatus *WSStatus*)

Parameters

sStoreNumber

User's store number.

sUser

User's login ID.

ipAddr

IP address of source machine (optional - used for logging).

Pass "" when address logging is not required.

WSStatus

UCAWebServiceStatus class describes the status of the call.

Return Values

Returns a value of 1 when successful.

Remarks

Any errors are logged to the NT Event log.

UpdateDeviceUser

This method updates device user information in the UCA server system. The user's information is updated in the B_USER table in the UCA database.

int UpdateDeviceUser(*UserInfo* newUserInfo, string *ipAddr*, ref UCAWebServiceStatus *WSStatus*)

Parameters

UserInfo

UserInfo class that describes the user (described above).

ipAddr

IP address of source machine (optional - used for logging).

Pass "" when address logging is not required.

WSStatus

UCAWebServiceStatus class describes the status of the call.

Return Values

Returns a value of 1 when successful.

Remarks

Any errors are logged to the NT Event log.

SetDeviceUserPIN

This method sets a device user PIN in the UCA system. The user's information is updated in the B_USER table in the UCA database.

int SetDeviceUserPIN(string *sStoreNumber,* string *sUser*, string *sPIN*, string *ipAddr*,
ref UCAWebServiceStatus *WSStatus)*

Parameters

sStoreNumber

User's store number.

sUser

Reference to a user's login ID.

sPIN

User password.

ipAddr

IP address of source machine (optional - used for logging).

Pass "" when address logging is not required.

WSStatus

UCAWebServiceStatus class describes the status of the call.

Return Values

Returns a value of 1 when successful.

Remarks

Any errors are logged to the NT Event log.

GetDeviceUserData

This method retrieves user information from the UCA system. The user's information is updated in the UserInfo class passed to the method.

int GetDeviceUserData(string *sStoreNumber*, string *sUser*, string *ipAddr*, ref UserInfo userInfo, ref UCAWebServiceStatus *WSStatus*)

Parameters

sStoreNumber

User's store number.

sUser

User's login ID.

ipAddr

IP address of source machine (optional - used for logging). Pass "" when address logging is not required.

UserInfo

A reference to a UserInfo class that describes the user.

WSStatus

UCAWebServiceStatus class describes the status of the call.

Return Values

Returns a value of 1 when successful. When successful, UserInfo is populated with the specified user's information.

Remarks

Any errors are logged to the NT Event log.

AddConsoleUser

This method adds a UCA Management Console user to the UCA system. The user is added into the B_USER table in the UCA database.

int AddConsoleUser(ConsoleUserInfo *userInfo*, string *sPassword*, string *ipAddr*, ref UCAWebServiceStatus WSStatus)

Parameters

ConsoleUserInfo

ConsoleUserInfo class that describes the user.

sPassword

User password.

ipAddr

IP address of source machine (optional - used for logging). Pass "" if the address should not be logged.

WSStatus

UCAWebServiceStatus class describes the status of the call.

Return Values

Returns a value of 1 when successful.

Remarks

Any errors are logged to the NT Event log.

Related Classes

```
public class ConsoleUserInfo
{
  public ConsoleUserInfo(){}
  public string ExtUsrID;
  public string firstName;
  public string lastName;
  public string RoleToken;
  public string userStatus;
  public DateTime dateCreated;
  public DateTime dateModified;
  public DateTime dateLastLogin;
  }
```

ConsoleUserInfo Class Members

Table 11-13 *ConsoleUserInfo Class Members*

Class Member	Description
ExtUsrID	User login ID.
firstName	User first name.
lastName	User last name.
RoleToken	Token representing user role (Admin, Manager). This data is found in the B_Roles table.
userStatus	User status (e.g., Active, Suspended, Disabled). This data is found in the S_UserStatusType table.
dateCreated	Date/Time the user was created.
dateModified	Date/Time the user was last modified.
dateLastLogin	Date/Time of last login.

RemoveConsoleUser

This method removes a console user from the UCA system. The user is removed from the B_ConsoleUser table in the UCA database.

int RemoveConsoleUser(string *sUser*, string *ipAddr*, ref UCAWebServiceStatus *WSStatus*)

Parameters

sUser

User's login ID.

ipAddr

IP address of source machine (optional - used for logging). Pass "" when address logging is not required.

WSStatus

UCAWebServiceStatus class describes the status of the call.

Return Values

Returns a value of 1 when successful.

Remarks

Any errors are logged to the NT Event log.

UpdateConsoleUser

This method updates a console user's information in the UCA server system.

int UpdateConsoleUser(ConsoleUserInfo *newUserInfo*, string *ipAddr*, ref UCAWebServiceStatus *WSStatus*)

Parameters

newUserInfo

ConsoleUserInfo class that describes the user (see *Table 11-13 on page 11-37*).

ipAddr

IP address of the source machine (optional - used for logging). Pass "" when address logging is not required.

WSStatus

UCAWebServiceStatus class describes the status of the call.

Return Values

Returns a value of 1 when successful.

Remarks

Any errors are logged to the NT Event log.

GetConsoleUserData

This method retrieves a console user's information from the UCA system. The user's information is updated in the ConsoleUserInfo class passed to the method.

int GetConsoleUserData(string sUser, string ipAddr, ref ConsoleUserInfo userInfo, ref UCAWebServiceStatus WSStatus)

Parameters

sUser

User's login ID.

ipAddr

IP address of source machine (optional - used for logging). Pass "" when address logging is not required.

UserInfo

A reference to a ConsoleUserInfo class that describes the user.

WSStatus

UCAWebServiceStatus class describes the status of the call.

Return Values

Returns a value of 1 when successful. When successful, UserInfo is populated with the specified user's information.

Remarks

Any errors are logged to the NT Event log.

SetConsoleUserPassword

This method sets a console user's password in the UCA server system.

Int SetConsoleUserPassword(string sUser, string sPassword, string ipAddr, ref UCAWebServiceStatus WSStatus)

Parameters

sUser

Reference to a user's login ID.

sPassword

User password.

ipAddr

IP address of the source machine (optional - used for logging). Pass "" when address logging is not required.

WSStatus

UCAWebServiceStatus class describes the status of the call.

Return Values

Returns a value of 1 when successful.

Remarks

Any errors are logged to the NT Event log.

Console User Authentication Web Service

In the default installation, the Console User Authentication Web Service is located at:

http://UCAInstallMachine/Motorola/UCA/ServerManagement/Authentication/ManagementAuthentication.asmx.

Console User Authentication Web Service Methods

GetUserData

This method retrieves user information from the UCA system. The user's information is updated in the ConsoleUserInfo class passed to the method.

int GetUserData(string *sUser*, string *ipAddr*, ref Console*UserInfo* userInfo, ref UCAWebServiceStatus *WSStatus)*

Parameters

sUser

User's login ID.

ipAddr

IP address of source machine (optional - used for logging).

Pass "" when address logging is not required.

UserInfo

A reference to a ConsoleUserInfo class that describes the user (described above).

WSStatus

UCAWebServiceStatus class describes the status of the call.

Return Values

Returns a value of 1 when successful. When successful, UserInfo is populated with the specified user's information.

Remarks

Any errors are logged to the NT Event log.

AuthenticateUserPassword

This function attempts to authenticate a UCA console user and password. It only checks the validity of the username and password pair.

int AuthenticateUserPassword(string *sUser*, string *sPassword*, string *ipAddr*, ref UCAWebServiceStatus *WSStatus*)

Parameters

sUser

User login ID.

sPassword

User password.

ipAddr

IP address of source machine (optional - used for logging).

Pass "" when address logging is not required.

WSStatus

UCAWebServiceStatus class describes the status of the call

Return Values

Returns a value of 1 when successful. If the return value is 0, check WSStatus for the error code and description.

Remarks

Any errors are logged to the NT Event log.

LoginConsoleUser

This function attempts to authenticate a UCA console user and password. If authenticated, the console user's records in the UCA database are updated.

int LoginConsoleUser(string *sUser*, string *sPassword*, string *ipAddr*, ref UCAWebServiceStatus *WSStatus*)

Parameters

sUser

User login ID.

sPassword

User password.

ipAddr

IP address of source machine (optional - used for logging). Pass "" when address logging is not required.

WSStatus

UCAWebServiceStatus class describes the status of the call.

Return Values

Returns a value of 1 when successful. If the return value is 0, check WSStatus for the error code and description.

Remarks

Any errors are logged to the NT Event log.

Customizing the Console Login Process

The default UCA Console User Authentication Web Service can be replaced by an Authentication Web Service written by a customer or integrator.

To replace the default UCA Console User Authentication Web Service with a customized Authentication Web Service:

1. Create a Console User Authentication Web Service that matches the signature, methods and return values of the UCA Console User Authentication Web Service.
2. Change the UCAConsole Web.config Web reference to point to the new Console User Authentication Web Service.

Motorola provides an example Console User Authentication Web Service (SampleConsoleAuth) with method stubs that can be used as a starting point for a customer or integrator creating a customized version.

To change the Web.config to use a different Authentication Web Service, find the following section in the Web.config file:

```
<applicationSettings>
  <UCACommon.Properties.Settings>
    <setting name=" UCACommon_ConsoleAuthService_ManagementAuthentication" serializeAs="String">
    <value>http://localhost/Motorola/UCA/ServerManagement/Authentication/
                ManagementAuthentication.asmx </value>
    </setting>
  </UCACommon.Properties.Settings>
</applicationSettings>
```

Change the <value> tag (above) to point to the new Authentication Web Service. The UCA server calls the new Authentication Web Service during the UCA Console Login Process. In the default install, the UCAConsole Web.config is located in the following directory: C:\Inetpub\wwwroot\Motorola\UCA\UCAConsole.

UCA Server Optional API Samples

Two sample programs that demonstrate using the UCA server optional APIs are provided:

- UCA Messaging Sample - Uses the optional API to send asynchronous messages to all or individual users.
- UCA Phone Extension Example - Uses the optional API to add a new dynamic UCA phone extension to the UCA database.

UCA Messaging Sample

The compiled application installs along with UCA Server installation. When installed, the location of the UCA Messaging Sample API is: http:/UCAInstallMachine/Motorola/UCA/Samples/UCAMessaging/default2.aspx.

This sample API uses the DeviceMessaging Web Service and allows the developer to send a message (text) and specify an audio file (.wav) to be played on the device.

NOTE The specified audio file must be present on the device.

The message can be sent to a specific user, or all users.

When installed, the sample source code for this sample can be found in the directory:
\Program Files\Motorola\UCA Server\DevelopmentToolkit\Messaging.

UCA Messaging Screen

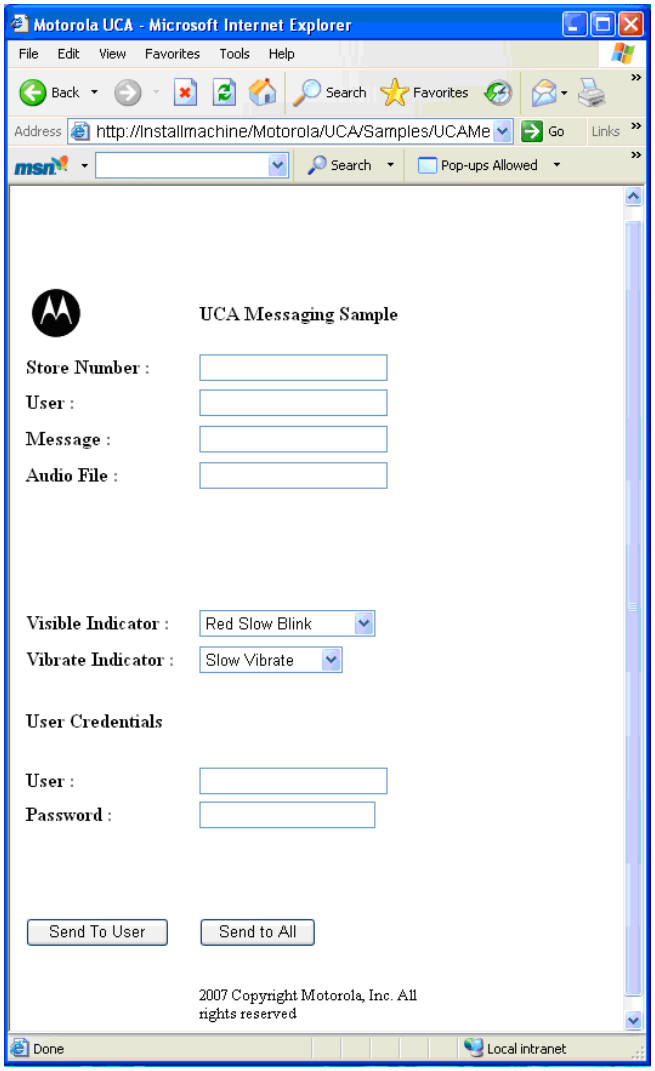


Figure 11-2 *UCA Messaging Screen Sample.*

Table 11-14 *UCA Messaging Sample Application - Screen Fields*

Field	Description
Store Number	The default store number is "0002."
User	User ID of a logged in CA50 device user. This field can be blank if sending the message to all users.
Message	Message text to display on the device.
Audio File	Path of the .wav file to play on the device. Leave this field blank if audio is not applicable.
User/Password	Valid user name and password (Windows login) for the user of the UCA server machine. The optional API is located in a protected directory on the machine.

Table 11-14 *UCA Messaging Sample Application - Screen Fields (Continued)*

Field	Description
Visible Indicator	Lights, color and blink rate to display.
Vibrate Indicator	Vibrate type.

UCA Phone Extension Sample

The compiled application is installed when *Developer Install* is selected at installation. If *Developer Install* is installed, the location of the UCA Phone Extension Sample API is: http:/UCAInstallMachine/Motorola/UCA/Samples/UCAPhoneExtension/default.aspx.

This sample uses the PhoneExtensionManagement Web Service and allows the developer to add a dynamically assigned UCA phone extension into the UCA database. When *Developer Install* is selected at installation, the sample source code for this API can be found in the directory:
\Program Files\Motorola\UCA Server\DevelopmentToolkit\PhoneExtensions.

UCA Phone Extension Screen

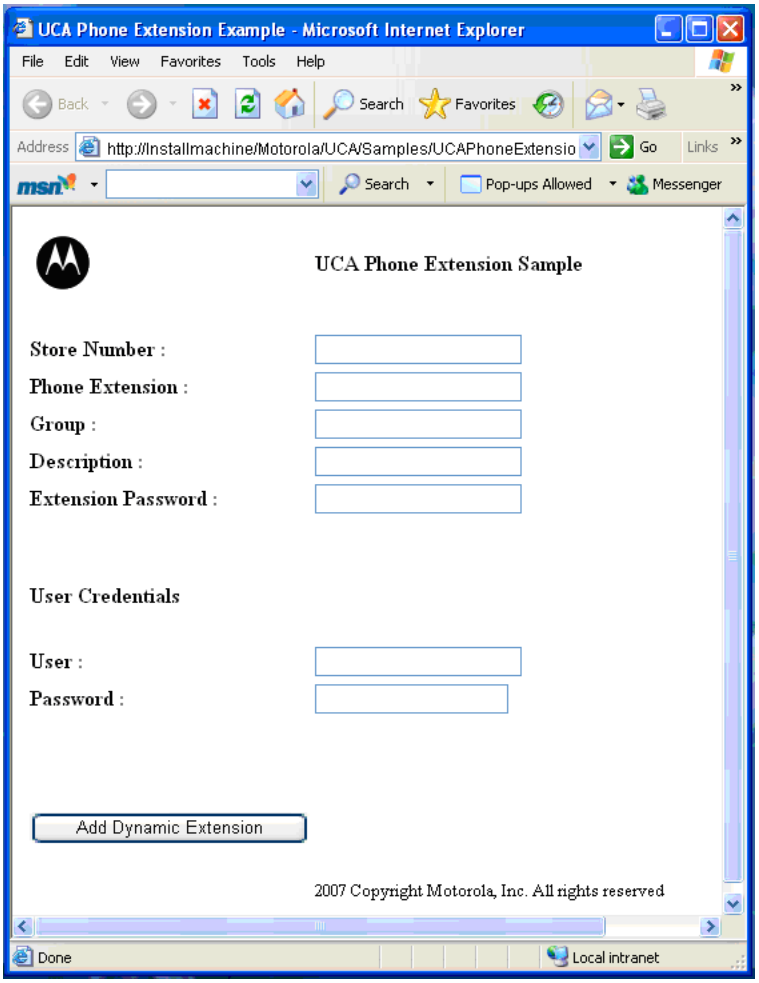


Figure 11-3 *UCA Phone Extension Screen Sample.*

Table 11-15 *UCA Messaging Sample Application - Screen Fields*

Field	Description
Store Number	The default store number is "0002."
Phone Extension	Extension number to add.
Group	Organizational group to which the extension should be assigned.
Description	Description of extension.
Extension Password	Password for extension with the PBX.
User/Password	Valid user name and password (Windows login) for the user of the UCA server machine. The optional API is located in a protected directory on the machine.

(Page intentionally blank.)

Chapter 12 Staging and Deploying CA50s

Overview

To stage and deploy CA50s at a customer site, use MSP Stage, which establishes communication with the RF infrastructure and initial connectivity to the UCA server.

To upgrade the CA50 OS, UCA client application or other software components use MSP Provision functionality.

One copy of the MSP 3.x server software (stage and provision functionality only) and its license certificate is provided free with the purchase of a UCA server software license.

- MSP 3.x license fees, per CA50, are free of charge.
  - CA50s are automatically registered with an MSP server upon initial staging (automated process, no user intervention required).
  - CA50s are automatically device licensed/enabled for stage and provision capabilities out-of-box (automated process, no user intervention required).
- Number of CA50s supported per MSP server - no software constraint/limitations are imposed.

NOTE If desired, an MSP Software Support (extended warranty) agreement must be purchased separately and in addition to any desired CA50 Server Software Support (extended warranty) agreement.

NOTE 1. For a complete checklist outlining all the development and deployment activities required to implement a CA50 solution, see the *Development and Deployment Checklist on page 2-1*.

2. For a table listing the most commonly requested topics within this manual, see the Quick Startup Guide on the last page of the manual.

Staging and Deploying CA50s

Rapid Deployment Bar Codes

Rapid Deployment bar codes generated by MSP include staging information, such as wireless network information, Store Number and UCA server IP address. When Rapid Deployment completes the device configuration process, the device has a valid network configuration and the UCA client application communicates with a UCA server.

CA50 Parameters Required to Generate RD Bar Codes

Before generating Rapid Deployment bar codes using MSP, the CA50 deployment parameters that follow must be defined. These deployment parameters to set are stored in the UCA.settings.xml object.

- UCA Server IP address
- Store Number
- SSL (True/False)
- Ignore SSL Certificate Errors (True/False)
- HTTP Port
- HTTPS Port
- Persist Settings (True/False)
- Web Service Path - Default Path Specified
- Logoff Service Path - Default Path Specified
- New Phone Service Name - Default Path Specified
- Session Check Service Name - Default Path Specified
- Language - English specified as default (available from MSP 3.2.1 onwards)

For details about how to access and launch the CA50 Rapid Deployment client, see *Table 4-6 on page 4-19* and *Appendix F, Rapid Deployment*.

UCA.settings.xml File for Configuring the UCA Client Application

UCA.settings.xml is an MSP settings profile that allows the user to configure the UCA client application on the CA50. Its settings consist of the server IP address, the store number, SSL configuration and a few Web service settings. The user can use these settings to generate RD bar codes so the UCA client application is able to communicate with the UCA server. These settings from the UCA.settings.xml are defined as follows.

The SSL (Secured Socket Layer) entry is used to enable or disable the use of SSL for Web Service calls and communication to and from the Web server. A value of 1 enables the use of SSL and a value of 0 disables the use of SSL. This entry resides in UCAPlugInPersist.reg.

The IgnoreSSLCerErrors informs the WININET layer whether or not to ignore SSL/PCT-based-certificates errors when using SSL for Web service calls or communication to and from the Web server. This entry resides in UCAPlugInPersist.reg.

The HTTPPort entry is the port used for WebService or Web server communication when SSL is not enabled. This entry resides in UCAPlugInPersist.reg.

The HTTPSPort entry is the port that is used for the WebService or Web server communication when SSL is enabled. This entry resides in UCAPlugInPersist.reg.

The LogOffService entry contains the name of the Web service that provides the "Log Off" operation. This name should not change but a new value can be stored here if for any reason the Web service name changes for the Log Off operation. This Web service operation is used when the CA50 client application is informed to log off the user. This entry resides in UCAPlugInPersist.reg.

The NewExtensionService entry contains the name of the Web service that provides the "request new phone extension" operation. This name should not change but a new value can be stored here if for any reason the Web Service name changes for the "request new phone extension" operation. This Web Service operation is used by the CA50 client application when it is necessary to obtain a new phone extension. This entry resides in UCAPlugInPersist.reg.

The SessionCheckService entry contains the name of the Web service that provides the "client checks in with UCA server (ServerCheckIn)" operation. This name should not change but a new value can be stored here if for any reason the Web service name changes for the "ServerCheckIn" operation. The CA50 client application checks in with the UCA server if it is determined that the CA50 device went out of range and then back in range. This entry resides in UCAPlugInPersist.reg.

The ServerIP entry contains the IP address of the UCA server. This entry resides in UCAPlugInPersist.reg.

The StoreID entry contains the store number of the store and is associated with the CA50 device. This entry resides in UCAPlugInPersist.reg.

Note that there are other UCAClient application registry definitions that are not necessary to stage that reside in UCAClient.reg. These settings can be changed manually or via a package that can be downloaded when deemed necessary. The additional settings are as follows.

The LoggingEnabled entry allows a yes or no toggle for logging. A value of 1 means logging is on. A value of 0 means logging is off. If logging is on, the DebugLevel keys allow more control of what gets logged at the component level. The higher the log level, the more log messages get printed out. These entries reside in UCAClient.reg.

The WebServiceResponseTimeout entry informs the CA50 client application how long to wait for a response from the server in seconds when attempting to make a Web Service call. The WebSericeTimeBeforeWarning is the amount of time to wait before displaying its *Waiting For Server* screen when waiting for a response from a Web service call. This entry resides in UCAClient.reg.

The WebSericeTimeBeforeWarning is the amount of time to wait before displaying its *Waiting For Server* or *Waiting for Web Server* screen when waiting for a response from a Web service call or Web server, respectfully. This entry resides in UCAClient.reg.

The LocalProfileFile entry is used if a profile is created and stored on the device so that the CA50 client application can run without a UCA server. However, the profile would have to be specifically made to run without the UCA server. This may be useful if the user only wishes to use the walkie-talkie and business applications. This entry resides in UCAClient.reg.

The LogFileLocation entry is used to inform the CA50 client application where to write the log file if logging is enabled. This entry resides in UCAClient.reg.

The language entry informs the CA50 client application which language the strings on the UCA client are to be shown.The supported languages are English, French, Italian, German and Spanish.The language entry resides in the UCAClient.reg. Please ensure that the languages selected from MSP for the client and sever (selected during the installation from install shield) match.

Staging Process for a CA50

The following steps explain how to stage the CA50 so that it enables the MSP, allows universal conditions, contains persistent network settings, relay server settings, UCA client settings and warm boot the device. Upon reboot the UCAClient application launches since its UCA settings are stored in the registry.

1. **Creating the Agent 30 settings definition**
   - In a Web browser, go to http://[server-ip]/msp.web to log in to an MSP server.
   - Select the *Library* tab.
   - Select the *Settings* tab.
   - Select the **Create** button.
   - Select the *Agent.30.setting.xml*.
   - Enter *Enable30* for the name field.
   - Select the **Next** button.
   - Enter *Enable 30 Agent* in the description field.
   - Enter *Enable30* for the name field.
   - Set the schedule interval to a time that is applicable to your environment. This time represents how often the MSP agent checks for MSP downloads (MSP jobs). It is recommended to check in every 15 minutes.
   - Set the update discovery document interval (Force regDoc Uploads) to a time that is applicable to your environment. It is suggested to set this to a value of 32 so that the discovery document is sent back to the MSP server every 8 hours (32 * 15 minutes) if it changed.
   - Ensure the *Enable30Agent* field is set to TRUE.
   - Click the **Finish**.
   - If the MSP server version is earlier than 3.1.1, the settings definition file should be downloaded from: http://support.symbol.com/support/product/softwaredownloads.do.
2. **Creating the UCA settings definition**
   - In a Web browser, go to http://[server-ip]/msp.web to log in to an MSP server.
   - Select the *Library* tab.
   - Select the *Settings* tab.
   - Select the **Create** button.
   - Select the *UCA.settings.xml*.
   - Enter *UCA Client Setup* for the name field.
   - Enter *UCA Client Setup* in the description field.
   - Enter the server IP address of the UCA server UCA Client Setup.
   - Enter the store number in which the UCA server is installed.
   - Enter the desired SSL entry depending on whether or not SSL should be enabled or disabled.
   - Specify whether or not the SSL certificate errors should be ignored.
   - Enter the HTTP port when SSL is not used or use the default.

- Enter the secured HTTP port when SSL is used or use the default.
- Use the default for the *Log Off Service*.
- Use the default for the *New Extension Service*.
- Use the default for the *Session Check Service*.
- Select Language, use the default web service path.
- Select *TRUE* for persist on a cold boot.
- Click the **Finish** button.
- This UCA settings definition only applies to the UCA client application running on the CA50.
- If the MSP server version is earlier than 3.1.1, the settings definition file should be downloaded from: http://support.symbol.com/support/product/softwaredownloads.do.

3. **Creating an MSP bundle**
   - In a Web browser, go to http://[server-ip]/msp.web to log in to an MSP server.
   - Select the *Bundles* tab.
   - Click the **Create** button.
   - Enter *StageCA50* for the *Name*.
   - Enter *Stage CA50* for the *Description*.
   - Click the **Next** button.
   - Click the **Add Step** button.
   - Select the user defined package type.
   - Select *the latest Universal package type (for Ex: Universal 1.1)* for the package type.
   - Click the **Add Bundle Step** button.
   - Click the **Add Step** button.
   - Click the **Content-Setting** button.
   - Select *Agent 30-EnableAgen30*.
   - Click the **Add Bundle Step** button.
   - Select *Add Step*.
   - Click the **Reboot** button.
   - Select *Warm* for *Type of Reboot* (see *Auto-launching the UCA Client Application* for additional information).
   - Click the **Add Bundle Step** button.
   - Click the **Finish** button.
4. **Creating an MSP staging profile**
   - In a Web browser, go to http://[server-ip]/msp.web to log in to an MSP server.
   - Select the *Staging* tab
   - Select the *Create* tab.
   - Enter *StageTheCA50* in the *Name* field.

- Click the **Next** button.
- Select the appropriate Network Access Setting for your environment where this network access setting must be configured to persist across a cold boot.
- Select the appropriate *Relay Server* for your environment.
- Select the *UCA Client Setup-UCA.setting.xml* for the additional setting.
- Click the **Next** button.
- Click the **User-Defined** button.
- Select *Stage CA50* for the bundle.
- Click the **Next** button.
- Deselect any bar code configuration you do not wish to use.
- Click the **Finish** button.

Refer to MSP documentation to print RD bar codes from the staging profile called *Stage the CA50*. Refer to *Appendix F, Rapid Deployment* to use Rapid Deployment on the CA50.

Auto-launching the UCA Client Application

The CA50 can be programmed to automatically launch directly into the UCA client application with no user intervention by selecting reboot and setting the type of reboot to *Warm (*as directed in bullet 15 under *Creating an MSP bundle*).

NOTE For auto-launch to take effect, the entire staging process, through and including *Creating an MSP staging profile*, must be completed.

When the type of reboot is set to *Warm*, auto-launch into the UCA client application is activated in the following conditions:

- after scanning RD bar codes
- after performing a warm boot
- after performing a cold boot
- after removing/inserting a battery.

Staging and Localization

If the CA50 device is staged using MSP versions prior to MSP 3.2.1, the language option does not appear in the UCA settings and the CA50 device does not receive language information during the staging process. The CA50 login screen displays in English only. After the device logs in, all display switch to the language set during the installation of UCA server software.

OS/UCA Client Application Upgrades Pushed from MSP Provision

When using MSP to provision multiple CA50s in close proximity with a new operating system, saturation of the Access Point on the network must be taken into account.

Devices need to be in an area of excellent RF coverage, so additional infrastructure may need to be installed to enable the best coverage and data rate possible. In addition, the CA50 should be in its charging cradle and must be powered on. When not on AC power, AirBeam / OSUpdate checks battery capacity before updating the partitions. Battery capacity below 75% results in an *Insufficient Power* message. MSP Provision functionality can be initiated either manually by scanning Rapid Deployment bar codes or automatically using an Over-the-Air command.

By default, from the factory, the CA50 MSP agent is not enabled. The MSP agent running on the CA50 provides a means to automate the updating of CA50 software among other features.To enable the MSP agent running on the CA50 generate the appropriate Rapid Deployment bar code(s). For details about how to generate Rapid Deployment bar codes, refer to the MSP user manual.

Minimizing Network Traffic from the MSP Agent

To minimize the network traffic from CA50s checking for software updates on the MSP server, generate an MSP policy that enables the CA50 to check for MSP downloads (also called MSP jobs) only when no user is logged onto the device (UCA client application is running but the user is logged out). The MSP policy can only function properly if the CA50 MSP agent is enabled. Refer to the latest MSP documentation on how to create a policy with the universal condition and how to activate the policy.

MSP Setup on the CA50 for MSP Provisioning

Perform the following steps to perform periodic check-ins based on a universal condition.

1. **Creating the universal condition for the UCAClient application**
   - In a Web browser, go to http://[server-ip]/msp.web to log in to an MSP server.
   - Select the *Library* tab.
   - Select the *Conditions* tab.
   - Click the **Create** button.
   - Select *Universal.condition.xml* for the condition type.
   - Enter a name for the condition, such as *UCALoggedOff*.
   - Select the **Next** button.
   - Enter a description for the condition, such as *condition for UCA*.
   - Select *Constraint* based on the registry key or value existence for the *Constraint Type*.
   - Select *Test* for the existence of the selected item for the *Test Condition*.
   - Select *HKEY_LOCAL_MACHINE* for the *Registry Base Key.*
   - Enter *Software\Motorola\UCAClient\LoggedOff* for the *Registry Key* or *Value Name*. In this case it is a registry key because of the '\' after *LoggedOff.*
   - Click the **Finish** button.
2. **Policies with the universal condition for the UCAClient application**

Any policies created with this condition cause the MSP client to only perform a policy when a user is logged out. The UCAClient application removes the universal condition registry key defined above when a user is logged in. The UCAClient application adds the universal condition registry key defined above when a user is logged out. This in turn allows you to instruct the MSP client to process the policy.

(Page intentionally blank.)

Chapter 13 Site Installation Guidelines

Overview

This chapter provides information for planning and preparing the UCA server and CA50 client for proper customer deployment.

Valuable topics in this chapter include:

- UCA server requirements (*page 13-7*)
- Stages of planning and preparation (*page 13-8*)
- Pre-deployment activities - first site (*page 13-8*)
- Deployment activities - per site (*page 13-8*)
- Planning and preparation activities (*page 13-9*)
- Developing a list of roles (*page 13-9*)
- Developing a list of groups (*page 13-9*)
- Planning walkie-talkie channels and addresses (*page 13-9*)
- Planning user authentication - device users and console users (*page 13-9*)
- Planning PBX integration (*page 13-10*)
- Developing user profile templates and associations (*page 13-10*)
- Developing custom business applications - optional (*page 13-10*)
- Identifying site specific settings (*page 13-10*)
- Configuring site user lists with role/group assignments (*page 13-10*)
- Configuring a site phone extension pool (*page 13-10*)
- Configuring additional phone book entries (*page 13-10*).

Site Installation Guidelines Overview

The planning and preparation for customer deployment is a critical step in optimizing the implementation and utilizing details of the UCA server and the UCA client application on the CA50. It involves setting up the server to meet the exact needs of the customer. A server set up can range from basic configurations that include required customer settings and options to developing various custom business applications, a customer specific user authentication Web service and a PBX integration service.

The Motorola UCA product requires information about a site environment to be seeded into the UCA server at installation time. This information is used to organize device users into phone book groups, identify statically assigned phone extensions, walkie-talkie channels, etc. The accuracy of this data must also be maintained over time as the site environment changes. Certain types of site changes can have ripple effects on multiple types of data. For example, new departments could be added which would affect the list of groups, device users, phone extension pool and additional phone book entries.

NOTE 1. For a complete checklist outlining all the development and deployment activities required to implement a CA50 solution, see the *Development and Deployment Checklist on page 2-1*.

2. For a table listing the most commonly requested topics within this manual, see the Quick Startup Guide on the last page of the manual.

Maximum Number of CA50s Deployed Per Site

The number of CA50s that can be deployed within a site is dependent on the site/installation specifics:

- Number of access points within the store
- Mode and rate of activity of CA50 usage
  - Voice (unicast) calls
  - Walkie-talkie communication
  - Bar code scanning
  - Messaging/paging
- Server hardware specifics
- Wired network infrastructure capability
- Other devices competing for network bandwidth.

Wireless Network Optimization

Maximizing Battery Life via Wireless Optimizations

For recommendations see *Table 3-1 on page 3-9*.

For recommendations on optimizing for walkie-talkie communication see *page 4-31*.

Decrease Roam Times Through a Channel Mask

To decrease roam times the CA50 ships from the factory with a channel mask implemented. The channel mask, limits the scanned RF channels to a subset of the total available channels.

By default, the CA50 ships with a channel mask enabling the use of only channels 1, 6 and 11.

Total Available Channels

Table 13-1 lists the available 802.11 a/b/g channels and the frequencies, not limited by regional constraints.

NOTE This table lists all available channels and does not specify which channels may be illegal to use based on regional/national laws.

Table 13-1 *Channel Availability*

Channel	Frequency
1	2412
2	2417
3	2422
4	2427
5	2432
6	2437
7	2442
8	2447
9	2452
10	2457
11	2462
12	2467
13	2472
14	2484

Allowed Channel by Region

Factory Default Channel Mask Reg Key - Channels 1, 6 and 11

North American Domain Channel Scheme - Channels 1 through 11

ESTI Domain Channel Scheme - Channels 1 through 13

Japan Channel Scheme - Channels 1 through 14

Table 13-2 *Allowable Channels, Users and Maximum Power Levels**

Channel	Frequency (MHz)	United States 20 MHz[8]	Europe 20 MHz	Japan**		Singapore 20 MHz	China 20 MHz	Israel 20 MHz[5]	Korea 20 MHz[10]
				20 MHz[9]	10 MHz				
7	5035	No	No	No	Yes	No	No	No	No
8	5040	No	No	No	Yes	No	No	No	No
9	5045	No	No	No	Yes	No	No	No	No
11	5055	No	No	No	Yes	No	No	No	No
12	5060	No	No	No	No	No	No	No	No
16	5080	No	No	No	No	No	No	No	No
34	5170	No	No	No	No	No	No	Yes	Yes
36	5180	Yes	Yes	Yes	No	Yes	No	Yes	Yes
38	5190	No	No	No	No	No	No	Yes	Yes
40	5200	Yes	Yes	Yes	No	Yes	No	Yes	Yes
42	5210	No	No	No	No	No	No	Yes	Yes
44	5220	Yes	Yes	Yes	No	Yes	No	Yes	Yes
46	5230	No	No	No	No	No	No	Yes	Yes
48	5240	Yes	Yes	Yes	No	No	No	Yes	Yes
52	5260	Yes	Yes	Yes	No	No	No	Yes	Yes
56	5280	Yes	Yes	Yes	No	No	No	Yes	Yes
60	5300	Yes	Yes	Yes	No	No	No	Yes	Yes
64	5320	Yes	Yes	Yes	No	No	No	Yes	Yes
100	5500	Yes	Yes	Yes	No	No	No	No	Yes
104	5520	Yes	Yes	Yes	No	No	No	No	Yes
108	5540	Yes	Yes	Yes	No	No	No	No	Yes
112	5560	Yes	Yes	Yes	No	No	No	No	Yes
116	5580	Yes	Yes	Yes	No	No	No	No	Yes
120	5600	Yes	Yes	Yes	No	No	No	No	Yes
124	5620	Yes	Yes	Yes	No	No	No	No	Yes
128	5640	Yes	Yes	Yes	No	No	No	No	Yes
132	5660	Yes	Yes	Yes	No	No	No	No	No

Table 13-2 *Allowable Channels, Users and Maximum Power Levels* (Continued)*

Channel	Frequency (MHz)	United States 20 MHz[8]	Europe 20 MHz	Japan**		Singapore 20 MHz	China 20 MHz	Israel 20 MHz[5]	Korea 20 MHz[10]
				20 MHz[9]	10 MHz				
136	5680	Yes	Yes	Yes	No	No	No	No	No
140	5700	Yes	Yes	Yes	No	No	No	No	No
149	5745	Yes	No	No	No	Yes	Yes	No	Yes
153	5765	Yes	No	No	No	Yes	Yes	No	Yes
157	5785	Yes	No	No	No	Yes	Yes	No	Yes
161	5805	Yes	No	No	No	Yes	Yes	No	Yes
165	5825	Yes	No	No	No	Yes	Yes	No	Yes
183	4915	No	No	No	Yes	No	No	No	No
184	4920	No	No	Yes	Yes	No	No	No	No
185	4925	No	No	No	Yes	No	No	No	No
187	4935	No	No	No	Yes	No	No	No	No
188	4940	No	No	Yes	Yes	No	No	No	No
189	4945	No	No	No	Yes	No	No	No	No
192	4960	No	No	Yes	No	No	No	No	No
196	4980	No	No	Yes	No	No	No	No	No

***Consult the local authorities, these regulations are subject to change at any time.**

**** The Japanese channels operating at 10 MHz bandwidth operate at half the data rate as the channels with 20 MHz bandwidth for the same modulation scheme.**

[5], [8], [9], [10] **See *Step 8. RF Network on page D-16*.**

Changing the Channel Mask

To change the channel mask:

1. Create a new reg key with the desired frequencies of the channels to scan. Name the reg key, channelmask.reg.
2. Put this reg key into the CA50 \Application folder. For information about doing this in a deployment, see *page 12-1*.
3. Cold boot the CA50 for the new mask to take effect.

Sample (suggested) channelmask.reg format

REGEDIT4

```
;###################### Channel Mask 1-6-11


[HKEY_LOCAL_MACHINE\Comm\JEDI10_1\PARMS]
"ChannelMask"="2412 2437 2462"


[HKEY_LOCAL_MACHINE\Comm\PW\JEDI10_1\PARMS]
"ChannelMask"="2412 2437 2462"
```

Battery Management

Maximizing Battery Life via Wireless Optimizations

For recommendations see *Table 3-1 on page 3-9*.

For recommendations on optimizing for walkie-talkie communication see *page 4-31*.

UCA Server Requirements

To set up the UCA server, prepare the information listed in each category below for each store environment.

Phone Information

The information below is required to present a phone book on the device that contains UCA users and callable non-UCA phones and phone services.

- Phone extensions at a site that are available for assignment to a CA50.
- Group (Departments) to which each UCA phone extension belongs.
- Non-UCA phones in the store that can be called by a CA50.
- Group (Departments) to which each non-UCA phone belong.
- Special dial strings that may be needed by the users as phone book entries (i.e., paging).
- Associates using CA50s that need the ability to call other sites.

Operations Information

- Groups (Departments) present at this site.
- Active roles at this site (e.g., Manager, Sales, Front End, etc.).
- Store number.

Walkie-Talkie Information

- Walkie-talkie channels and their definitions at this site.

IT Information

- IP addresses and range of ports to use at this site for the multicast IP traffic used by the walkie-talkie functionality.
- IP address (or domain name) and port of the PBX.
- IP address (or domain name) of the UCA server.

Stages of Planning and Preparation

Pre-deployment Activities - First Site

There are several integration steps to take before deploying the first site.

1. Develop a company-wide list of all *Roles*.
2. Develop a company-wide list of all *Groups* (Departments).
3. Plan walkie-talkie channel allocations and multicast addresses/ports for the walkie-talkie channels.
4. Plan user authentication (*Device Users* and *Console Users*).
5. Plan PBX integration.
6. Develop *User Profile* templates and associations.
7. Develop custom business application(s). This is optional.

Deployment Activities - Per Site

Prior to deploying each UCA system at a site, prepare the following for each site:

- Identify site specific settings, such as UCA server address, PBX address, etc.
- Configure a user list with *Role* and *Group* assignments for the site.
- Configure a phone extension pool for the site.
- Configure additional phone book entries for the site.

Planning and Preparation Activities

Developing a List of Roles

To develop a list of company-wide roles, use associate titles. The most important aspect of a role is selecting a profile template that allows each role to a range of privileges. For example, some associates might have access to business applications where other associates may not.

Developing a List of Groups

To develop a list of company-wide groups, or departments, use phone extension pools and additional phone book entries. The groups drive how a phone extension gets assigned when an associate logs in. It is then used to organize the phone book to facilitate looking up various associates.

Planning Walkie-Talkie Channels and Addresses

Plan walkie-talkie channel allocations and multicast addresses/ports for the walkie-talkie channels. Walkie-talkie channels use multicast packets and this requires a plan when using multicast addresses to route the multicast packets. Normally, channels are allocated on the same multicast address with incrementing port numbers.

NOTE The UCA server and the UCA client application reserve the first port for asynchronous communications from the server to the UCA clients. This means that Channel 1 on a device is not the first port in the range. It is recommended to start the port ranges on a '0'. For example, the multicast channels start on address 239.192.2.2 port 5000. This means that the reserved server communications channel is on port 5000. This means that walkie-talkie channel 1 is on port 5001, channel 2 is on 5002, and so forth.

Planning User Authentication (Device Users and Console Users)

The UCA server defaults to using the built in Device User Authentication and Console User Authentication via Web Services that access the UCA database. These authentication services can be redirected to alternate user authentication Web Services. If the built in user authentication is used, the user can be maintained via the Console or the User Management Web Service. If it is desired to use an external authentication database for either device users or console users, custom authentication Web Services can be developed based on the UCA Sample Authentication and UCA Console Sample Authentication reference implementations supplied in the UCA server Install. Because they are separate authentication Web Services, the Device Users and Console Users can be independently redirected.

For detailed information about the these optional APIs, see the following sections:

- *CA50 Device User Authentication Process on page 11-5*
- *Device User Authentication Web Service on page 11-8*
- *User Management Web Service on page 11-29*
- *Console User Authentication Web Service on page 11-40*.

	Built In Authentication Web Service	Built In Management Web Service
Device Users	UCA\Client\Authentication\Authentication.asmx	UCA\Server Management\UserManagement.asmx
Console Users	UCA\Server Management\Authentication\ ManagementAuthentication.asmx	UCA\Server Management\UserManagement.asmx

Planning PBX Integration

The CA50 must be assigned a phone extension. This requires the UCA server to be seeded with extensions that can be assigned to users at login time. In addition, the UCA server must be seeded with all phone numbers that an associate would need to call from a CA50.

Prepping the phone extension pool and additional phone book entries is a multi-step process:

1. Identify all extensions that are currently in use for desk phones, wall phones, etc.
2. Identify available extensions (new or unused) that can be placed into the phone extension pool.
3. Adjust call coverage groups and call rollover groups to include the extensions in the pool.
4. Associate the phone extensions in the pool with the appropriate groups for assignment when an associate logs in, and for display in the phone book.

NOTE An external application can be developed to use the Phone Extension Web Service to update the UCA server as part of the customer's process for maintaining the PBX.

Developing User Profile Templates and Associations

The user profile templates define access rights and privileges for various associates who use the CA50. Privileges are related to an associate's role and the primary association of a profile template is based on role.

Developing Custom Business Applications (optional)

To provide associates with additional capabilities such as price checking, item location, task management, etc., custom Web applications can be developed for the CA50 to access over the network. Sample price lookup applications are provided as examples in the UCA server install.

NOTE For sample business applications see *Appendix I, Business Applications Supplement*.

Identifying Site Specific Settings

As each site is deployed there are settings in the DB that are specific to the network structure at the site. This includes the UCA server address or host name, PBX address and URLs for any business applications used by the CA50. Depending upon the network characteristics there are some network timeout settings that may need to be adjusted. For more information, see the *Base XML Profile Sample on page 7-25* (WebServiceResponseTimeout, CacheTimeToLive, BadRFSampleThreshhold, GoodRFSampleThreshhold and MinSignalStrength).

Configuring Site User Lists with Role/Group Assignments

If the built in Device User Authentication is used, the tables must be seeded with the users at the site. Since the UCA login process depends upon a role and group for the associate logging in, an audit should be performed of the users at that site (either in the built authentication tables or in the external authentication database that was integrated).

Configuring a Site Phone Extension Pool

Phone extension setups and needs may vary from site to site. An audit is required to determine exactly how the PBX at the site is configured. Make any required adjustments to the phone extension pool and appropriately seed the phone extension pool.

Configuring Additional Phonebook Entries

Similar to the phone extension pool, additional phone book entries must be evaluated at each site to account for the variations at each site.

Chapter 14 Maintenance, Troubleshooting and Error Messages

Overview

This chapter includes instructions on cleaning the device and provides troubleshooting solutions for potential problems during device operation.

Valuable topics in this chapter include:

- Maintaining the UCA Server (*page 14-2*)
- Scheduled Data Maintenance (*page 14-2*)
- Keeping System Data and Configuration Up to Date (*page 14-3*)
- Maintaining the CA50 (*page 14-3*)
- Troubleshooting: UCA Server Installation/Uninstall (*page 14-4*)
- Troubleshooting: CA50 Client (*page 14-6*)
- Troubleshooting: CA50 Device (*page 14-11*)
- UCA Server System Error Code Number Index (*page 14-13*)
- UCA Server System Error Code Number Ranges (*page 14-16*)
- General Errors (*page 14-16*)
- UCA Database Errors (*page 14-17*)
- General Errors (*page 14-18*)
- UCA Database Errors (*page 14-34*).

✓ ***NOTE*** 1. For a complete checklist outlining all the development and deployment activities required to implement a CA50 solution, see the *Development and Deployment Checklist on page 1*.

2. For a table listing the most commonly requested topics within this manual, see the Quick Startup Guide on the last page of the manual.

Maintaining the UCA Server

Maintaining an installed UCA server requires:

- Scheduled data maintenance
- Keeping system data and configuration up to date.

Scheduled Data Maintenance

There are some data maintenance tasks that the UCA server periodically performs. It exports and purges debug trace data and metrics data. It also purges recent user records, device records and phone extension status records.

The UCA Maintenance Service manages this maintenance. It can run every day at the configured hour (system setting: System.Maintenance.DB.Hour). It can be configured to skip days according to the system setting: System.Maintenance.DB.DaysBetween.

Each area of maintenance has its own set of system settings:

- System.Maintenance.DB.DebugTrace
- System.Maintenance.DB.Metrics
- System.Maintenance.DB.Devices
- System.Maintenance.DB.LocalUser
- System.Maintenance.DB.PhoneExt
- System.Maintenance.DB.CallParkExt

Only the DebugTrace and Metrics generate export files. The files are written into the folder specified by System.Maintenance.DB.Export.Path. Both of these maintenance items has a FileAging setting. This setting controls how many automatic export files for this item are left in the export folder after it exports the current set of records.

All of these maintenance items have a DaysBetween setting. This controls how often the maintenance item runs. This is a threshold that is checked when the overall maintenance runs according to System.Maintenance.DB.DaysBetween. If the overall maintenance schedule is longer than one day, then any item set to more frequently than the overall schedule only runs according to the overall schedule.

All of these maintenance items also have a RecordAging setting. This setting controls how long a record is allowed to remain in the database before it is purged and optionally exported.

NOTE Devices and Local Users are recommended to be a relatively long interval, while Debug Trace is recommended to be set at daily.

Keeping System Data and Configuration Up to Date

Due to the change that a particular location is likely to undergo, there is a need to keep the system data and system configuration up to date. This varies from adding, removing, moving and renaming phone extensions to changing device users (new hires and employees that leave the company) to changing the PBX. There are several mechanisms available to keep this data up to date:

- Manual changes via the UCA Web Console.
- Importing a list of new phone extensions or device users via the UCA Web Console.
- Third party applications that use the UCA server optional APIs.

Manual Changes via the UCA Web Console

The UCA Web Console allows any necessary piece of data to be manually configured. Since the UCA Web Console is a manual process, this is most suitable for infrequently changing items. This is also intended as a user friendly way to understand what the various settings accomplish. (See *Chapter 9, UCA Server WebConsole Graphical User Interface* for more details.)

Importing a List of New Phone Extensions or Device Users via the UCA Web Console

Since initial setup of a server requires the entry of phone extensions and device users, a CSV import tool is provided. (See *Edit Phone Extension Window on page 12* and *All Users View on page 19* for more details.)

Third Party Applications That Use the UCA Server Optional APIs

The UCA server optional APIs are provided as a means to automate or allow alternate user interfaces to be developed to maintain the UCA system data. (See *Chapter 11, UCA Server Optional APIs* for more details.) This allows maintenance of related UCA data to be updated when the normal business process changes how the facility is arranged; be it phone extensions, groups or associates.

Maintaining the CA50

The degree of protection for the CA50 is specified by an ingress protection rating of IP 40 which is sufficient for retail applications but does not allow for the unit to be immersed in or soaked with liquid.

To clean the device a cloth or wipe moistened with alcohol is recommended. To disinfect the device standard sterilization wipes may be used.

Cleaning and/or Disinfecting

Wipes are the recommended methodology for cleaning the CA50 to avoid infiltration of liquid into the IP40 sealed housing. Cleaning the CA50 with a damp disinfecting wipe and allowing it to open air dry will not damage the unit. Prolonged soaking with a liquid agent may cause internal damage.

The following agents have been tested and approved:

- Sodium Hypochlorite
- Alkyl dimethyl benzyl ammonium chloride
- Ammonium Chloride
- Hydrogen Peroxide
- Isopropanol, Phosphoric Acid
- Isopropanol.

Troubleshooting

UCA Server Installation/Uninstall

Table 14-1 *Troubleshooting: UCA Server Install/Uninstall*

Symptom	Operating System	Cause	Solution
Cannot load server software.	Server 2003	Cannot load server software on a 2003 server, running active directory, that operates as a domain controller.	See *UCA Server Requirements and Setup on page 7*.
HTTP 404 Error Not Found when navigating to the Console Login page	Server 2003	Permission error.	Active server pages and ASP.NET v2.0.50727 must be set to allow for Web service extensions in IIS.
Waiting for Server Response, *Error Retry 10,9,8,7….seconds* displays on a CA50 with connectivity bars in the upper left display corner.	XP Pro SP2 Server 2003	Firewall on the UCA server is not turned off or CA50s cannot connect to the Server IP address through the wireless infrastructure.	Ensure firewall is turned off on the UCA server or allows CA50s to connect to the Server IP address through the wireless infrastructure. Try to ping UCA server IP address with another wireless device.
		Incorrect UCA server IP address setup using MSP 3.1 Stage Profile for RD Client.	Confirm that the correct UCA IP address is setup correctly in MSP 3.1.
Checking for updates... CA50 reboots, or the message displays every 15 seconds.	XP Pro SP2 Server 2003	MSP Client is not enabled on the device. A policy that contains an updated version of the UCA client application does not exist on the MSP server.	Verify the MSP Client is enabled on the device. Verify a policy exists on the MSP server that contains an updated version of the UCA client application.
During install, Error 27506 displays; password validation failed. The password does not meet Windows policy requirements because it is not complex enough.	XP Pro SP2 Server 2003	Password chosen for sa and/or database login is not strong (the password does not meet Windows policy requirements because it is not complex enough). The password must exceed eight characters and contain three of the following: uppercase, lowercase, numeric and non-alphanumeric.	Uninstall UCA instance of SQL Server 2005 from Add/Remove programs. Start UCA server install again using strong passwords.
Reinstalling on a target machine that had a previous version of the server installed: Error 27506. Error executing SQL script _uca_CreateDB.sql. Line 8. CREATE DATABASE failed. Some file names listed could not be created. Check previous errors.	XP Pro SP2 Server 2003	The database files UCA.mdf, UCA_log.ldf, SampleLookup.mdf and SampleLookup_log.ldf were not deleted from a previous install.	Uninstall the UCA instance of SQL Server 2005 from *Add/Remove* programs. Remove or relocate UCA and Sample Lookup databases and logs from: C:\Program Files\Microsoft SQL Server\MSSQL…2\MSSQL\DATA. Reinstall the UCA server

Table 14-1 *Troubleshooting: UCA Server Install/Uninstall (Continued)*

Symptom	Operating System	Cause	Solution
Reinstalling on a target machine that had a previous version of the server installed: Error 27506. Error executing SQL script _ucaSample_CreateDB.sql. Line 4. Cannot create file 'C:\Program Files\Microsoft SQL Server\MSSQL.1\MSSQL\Data\SampleLookup.mdb' because it already exists.	XP Pro SP2 and Server 2003	The database files SampleLookup.mdf and SampleLookup_log.ldf were not deleted from a previous install.	Uninstall the UCA instance of SQL Server 2005 from *Add/Remove* programs. Remove or relocate UCA and Sample Lookup databases and logs from: C:\Program Files\Microsoft SQL Server\MSSQL…2\MSSQL\DATA. Reinstall the UCA server.
Reinstalling on a target machine that had a previous version of the server installed: Error 27502 displays; could not connect to Microsoft SQL Server <hostname> Login failed for user "sa."	XP Pro SP2 Server 2003	UCA instance of the SQL server still resides on the target machine from a previous version of the UCA server.	Uninstall the UCA instance of SQL Server 2005 from *Add/Remove* programs. Remove or relocate UCA and Sample Lookup databases and logs from: C:\Program Files\Microsoft SQL Server\MSSQL…2\MSSQL\DATA. Reinstall the UCA server.
SQL Server 2005 install fails.	XP Pro SP2 and Server 2003	During install of SQL Server 2005, the sa password must meet SQL Server password policy requirements.	Cancel, then Next, then Finish and then cancel UCA install. Otherwise, after the remainder of the install fails with 27502 error, uninstall UCA instance of SQL Server 2005 from Add/Remove programs. Start UCA server install again using strong passwords.

CA50 Client

Table 14-2 *Troubleshooting the CA50 Client*

Problem	Cause	Solution
Error logging in.	Invalid bar code scanned. An error screen displays the following message on the CA50: *Invalid log in Try again*.	Scan another bar code to attempt log in again.
	User is already logged in. *Already logged in, log in here instead?* displays.	Respond “Yes” to close current user session and continue log in process. Respond “No” to return to log in screen.
Client does not turn on.	Lithium-ion battery not charged.	Charge or replace the lithium-ion battery in the device.
	Lithium-ion battery not installed properly.	Ensure battery is installed properly. See *Inserting and Removing the Battery on page 4*.
	System crash.	Perform a warm boot. If the device still does not turn on, perform a cold boot. See *Resetting the CA50 on page 17*.
Rechargeable lithium-ion battery did not charge.	Ambient battery charging temperature was too high or too low.	Move the cradle and device to another location with ambient temperatures between 0^{o} C and 35^{o} C. See *Charging Temperature on page 7*.
	Charge timer expired because the battery continuously charged for more than 4.5 hours. Expiration of the timer can be determined if the ambient temperature is within 0^{o} C to 35^{o} C accompanied by a blinking red LED and an "insert charge fault” icon on the display. Usable battery life may be expended.	If a battery repeatedly causes a timer expiration when charging from a dead or very low state, take battery out of service and replace it with a new battery.
	Battery failed.	Replace battery. If the device still does not operate, perform a warm boot, then a cold boot. See *Resetting the CA50 on page 17*.
	Device removed from cradle while battery was charging.	Insert device in cradle. The battery fully charges in less than three hours.
Cannot see characters on display.	Device not powered on.	Press any key to wake the device.
During data communication, no data transmitted, or transmitted data was incomplete.	Device disconnected from server during communication.	Contact technical support or the help desk to see if the network infrastructure is functioning properly. For example, device may have lost or did not obtain an IP address.
	Incorrect configuration.	See the system administrator.
CA50 does not emit sound.	Volume setting is low or turned off.	Adjust the volume. If this still does not solve the problem, contact the help desk.

Table 14-2 *Troubleshooting the CA50 Client (Continued)*

Problem	Cause	Solution
CA50 shuts off.	Battery is not inserted properly.	Insert the battery properly. See *Inserting and Removing the Battery on page 4*.
	Battery is depleted.	Replace the battery.
CA50 keeps powering down to protect memory contents.	The device battery is low.	Recharge the battery.
The CA50 does not accept scan input.	Scanning application is not loaded.	Load a scanning application on the device. See the system administrator.
	Unreadable bar code.	Ensure the symbol is not defaced.
	Distance between exit window and bar code is incorrect.	Place the device within proper scanning range.
	Device is not programmed for the bar code type.	Program the device to accept the type of bar code scanned.
	Device is not programmed to generate a beep.	If the device does not beep on a good decode, set the application to generate a beep on good decode.
	Battery is low.	If the scanner stops emitting a laser beam upon pressing the **Scan** key, check the battery level. When the battery is low, the scanner shuts off before the device low battery condition notification. Note: If the scanner is still not reading symbols, contact the distributor or the Motorola Global Customer Interaction Center.

Removing Invalid Entries from Add/Remove Programs.

If a program is uninstalled and the corresponding entry in the Add or Remove Programs listing was not removed. The listing may be manually removed from the Add or Remove Programs applet:

1. Click Start, Run and type regedit.exe
2. Navigate to: HKEY_LOCAL_MACHINE \ SOFTWARE \ Microsoft \ Windows \ CurrentVersion \Uninstall
3. Each subkey represents an installed program.
4. To determine which program that each key represents, click the key, and then view the *DisplayName* value data in the right pane.
5. Select the subkey to be deleted.
6. Right-click on the subkey and choose Delete.
7. Close Registry Editor

UCA_HOME Environment Variable Not Visible

After installing the UCA Server, log off and log on to see the changes.

Server Error in 'Motorola/UCA/UCAConsole' Application

Exception Details:

- System.NullReferenceException
- Object reference not set to an instance of an object.

One of the reasons for this error to occur is 'Encrypt.bat' script did not execute properly during installation

Solution:

1. Reboot the computer on which UCA Server software was installed.
2. Go to C:\Program Files\Motorola\UCA Server\Utilities.
3. Execute Encrypt.bat twice.

Logging File Provided by the UCAClient Application

The UCAClient application provides the means to generate a log file called UCAClient.txt in case there ever is a need to resolve an issue on the CA50 or on the UCA server. The contents of this log file are derived from two sources. The first source is the UCAClient.reg file which contains log settings that are imported into the CA50 registry. The second source is from the XML profile. The log settings from the UCAClient.reg file are used until a user profile is downloaded at which time the log settings are replaced from the Debug node in the user profile.

Log Settings in the UCAClient.reg File

Use ActiveSync, with a standard USB Mini-DIN cable, to access the UCAClient.reg and the log file. The default log file name is UCAClient.txt. The log settings contained in the UCAClient.reg file are as follows.

```
REGEDIT4

[HKEY_LOCAL_MACHINE\SOFTWARE\Motorola\UcaClient\Params]
     "LoggingEnabled"=dword:00000001
     "WTTDebugLevel"=dword:00000003
     "PhoneDebugLevel"=dword:00000003
     "ScannerDebugLevel"=dword:00000003
     "AgentDebugLevel"=dword:00000003
     "BrowserDebugLevel"=dword:00000003
     "LogFileLocation"="\UcaClientLog.txt"
```

The *LoggingEnabled* parameter value informs the UCAClient application to completely disable logging so no log file is generated. This parameter is the only mechanism to completely disable logging on the UCAClient application.

The *LogFileLocation* parameter value informs the UCAClient application the path and the name of the log file.

The *WTTDebugLevel* is the logging level for logic that is exercised to interface with the VoIP Walkie-Talkie subsystem component.

The *PhoneDebugLevel* is the logging level for logic that is exercised to interface with the VoIP Phone subsystem component.

The *ScannerDebugLevel* is the logging level for logic that is exercised to interface with the Scanner subsystem component.

The *AgentDebugLevel* is the logging level for logic that is exercised to interface with the Agent part of the UCAClient application. Basically, the Agent part is logic that controls the desktop window, menus, the user profile, and logic that communicates with each module that is responsible for interfacing to a particular subsystem component.

The *BrowserDebugLevel* is the logging level for logic that is exercised to process business applications.

The logging level for each of these can be one of five values explained below:

0: A value of 0 means that no logging will be provided.

1: A value of 1 means that only information messages will be logged in the log file.

2: A value of 2 means that information and error messages will be logged in the log file.

3: A value of 3 means that information, error, and warning messages will be logged in the log file.

4: A value of 4 means that information; error, warning and debug messages will be logged in the log file.

Log Settings in XML Profile (User Profile)

The log settings in the XML, or user profile, are contained in the XML node called *DebugLog* as illustrated below.

```
<DebugLog>
     <Log category="WTT" level="Off"/>
     <Log category="Phone" level="Info"/>
     <Log category="Agent" level="Warn"/>
     <Log category="Browser" level="Error"/>
     <Log category="Scanner" level="Debug"/>
</DebugLog>
```

The DebugLog node contains entries where each entry is a Log element. The Log element contains two attributes. The first attribute is the category. The category can contain one of five values. These values correspond to the log settings in the UCAclient.reg file as illustrated in *Table 14-3*.

Table 14-3 *Log Elements - Category Values*

Parameters in the UCAClient.reg File	Log Entry in the XML Profile
WTTDebugLevel	WTT
PhoneDebugLevel	Phone
ScannerDebugLevel	Scanner
AgentDebugLevel	Agent
BrowserDebugLevel	Browser

The second attribute of the Log entry is the level attribute. The level attribute informs the UCAClient application how much information is logged in the log file. The values for the level attribute also correspond to the values for the log settings in the UCACleint.reg file as illustrated in *Table 14-4*.

Table 14-4 *Log Elements - Level Attributes*

Value in the UCAClient.reg File for Debug Level Parameters	Value for the Level Attribute of the Log Entry in the XML Profile	Description
0	Off	No logging is provided.
1	Info	Only information messages are logged in the log file.
2	Error	Information and error messages are logged in the log file.
3	Warn	Information, error and warning messages are logged in the log file.
4	Debug	Information, error, warning and debug messages are logged in the log file.

NOTE The only way to disable the UCAClient application from generating a log file is to make the LoggingEnabled parameter a value of zero in the UCAClient.reg file and then reboot the CA50.

Log settings contained in the XML profile override log settings in the UCAClient.reg file when a user profile is received from the UCA server.

Example Log File Content

An example of the content of a log file generated by the UCAClient application is listed below.

05-03-2007, 05:44:15.56059 (000001ms), CAgent::SetComState wtmode=-1 ch=31

05-03-2007, 05:44:16.57137 (001078ms), cancel readnotify requestID= 1 notifpending=0 cancelpending=0

05-03-2007, 05:44:16.57341 (000204ms), CAudioAPI::SetVolumes SpeakerVolIndex = 3 ReceiverVolIndex = 3

05-03-2007, 05:44:16.57435 (000094ms), CVoIPPhone::InitPhone InitData=SERVERIP=192.168.0.2;LOCALIP=192.168.0.8;NAME=Anna Carter;EXT=221;PIN=1234;GATEKEEPER=Standard MicAtten=25 RcvAtten=80 RecvTone=\Application\wav\ringtone basic.wav

05-03-2007, 05:44:18.58921 (001486ms), CAgent::SetComState wtmode=0 ch=3

05-03-2007, 05:44:18.58923 (000002ms), CAgent::ActivateWindow ACTIVATED

05-03-2007, 05:44:18.58934 (000011ms), new readnotify requestID= 1 notifpending=0 cancelpending=0

05-03-2007, 05:44:18.58936 (000002ms), new readnotify requestidt= 3 result=0

05-03-2007, 05:44:23.64002 (005065ms), CVoIPWT::VoIPWTCallback WTRX Text Event: Channel= 3 text=<?xml version="1.0" encoding="utf-8"?><WTID Text="Abigail Jones" />05-03-2007, 05:44:23.64004 (000002ms), CVoIPWT::ProcessWndMessages WTRX Text Msg text=<?xml version="1.0" encoding="utf-8"?><WTID Text="Abigail Jones" /> rx sync id = 52308533 expecting sync id=52308533

05-03-2007, 05:44:32.08158 (009690ms), CVoIPWT::VoIPWTCallback WTRX Stop Event: Channel= 3 ID=0

05-03-2007, 05:44:51.26711 (018553ms), CAgent::ProcessWndMessages WM_KILLFOCUS active window=1

05-03-2007, 05:44:51.26719 (000008ms), cancel readnotify requestID= 3 notifpending=1 cancelpending=0

05-03-2007, 05:44:51.26730 (000011ms), read cancelled

05-03-2007, 05:44:52.27831 (001101ms), cancel readnotify requestID= -1 notifpending=0 cancelpending=0

CA50 Device

The CA50 contains several diagnostic applications that can be used to troubleshoot the CA50 device. *Table 14-5* includes possible problems and the applications to use to help troubleshoot.

Table 14-5 *Troubleshooting the CA50 Device*

Problem	Solution
The red laser beam does not emit when the **Scan** key is pressed to scan bar codes.	Use *ScanSamp* to test the scanner as follows. From the *CA Menu*, use the **Up** and **Down** arrow keys and select *Programs* > *ScanSamp*. Aim the scan window at a bar code and press **Scan/Select**. • If the red scan beam emits and decoded bar code data displays on the device, the business application used for scanning may be faulty. • If the red scan beam does not emit and the bar code is not decoded, there may be a problem with the device.
Voice is inaudible.	Use the *Audio Test App* to test the CA50 speaker, receiver and microphone as follows. From the *CA Menu*, use the **Up** and **Down** arrow keys and press **Scan/Select** to select *Programs* > *Audio Test App* and ensure *Output* is set to *Speaker* (press **Scan/Select** to toggle *Output* from Handset to Speaker). • Select *Programs* > *Audio Test App*> *Play Audio*> *Play...* to test the speaker and receiver. • Select *Programs* > *Audio Test App*> *Record Audio*> *Record...* to test the speaker, receiver and microphone. If sound is audible, the business application used for audio may be faulty. If sound is not audible, there may be a problem with the device.
The CA50 battery in the device does not charge.	From the *CA Menu*, use the **Up** and **Down** arrow keys and press **Scan/Select** to select *Programs* > *UCA Sys Info* to verify correct battery temperature, voltage and current statistics. If the temperature is > 35^{o} C, place the device in the charger and move it to a cooler place.
CA50 seems sluggish.	From the *CA Menu*, use the **Up** and **Down** arrow keys and press **Scan/Select** to select *Programs* > *UCA Sys Info* to check memory allocation problems or WLAN connection issues. If AVAIL RAM is significantly lower than 18996 or if AVAIL VIRT is significantly lower than 29120, reboot the device. If the condition persists, contact Motorola Enterprise Mobility support (see *Service Information on page xxi*).

Table 14-5 *Troubleshooting the CA50 Device (Continued)*

Problem	Solution
There is no WLAN connection.	1. From the *CA Menu*, use the **Up** and **Down** arrow keys and press **Scan/Select** to select *Settings* > *Network* > *Network Status ...* to check the *Active Profile* and *IP*. If there is no IP address, consider connecting to a different network. 2. From the *CA Menu*, use the **Up** and **Down** arrow keys and press **Scan/Select** to select *Programs* > *UCA Sys Info*. • Use the left soft key (*Next*) to display the *IP* screen. If there is no IP for the WLAN, then there are problems communicating with the radio. • Use the left soft key (*Next*) to display the *MAC* screen to see if there is a MAC for the radio. If the IP is 0.0.0.0, the radio cannot connect to an AP. If the IP is 0.0.0.0, verify that the *Active Profile* is correct and the AP is within range. 3. Verify that the country code of the AP matches the country code of the wireless profile. The 802.11d feature is enabled by default and prevents a connection to an AP with a different country code or with no country code.
Keys pressed on the CA50 do not operate.	From the *CA Menu*, use the **Up** and **Down** arrow keys and press **Scan/Select** to select *Programs* > *UCA KeyCheck*. Press the keys on the CA50 to see if a key code is generated by every key press. If key codes are not generated, contact Motorola Enterprise Mobility support (see *Service Information on page xxi*).
LEDs do not light or vibrator motor does not turn on.	From the *CA Menu*, use the **Up** and **Down** arrow keys and press **Scan/Select** to select *Programs* > *UCA NLedTest*. Test the red and green LEDs and the vibrator motor by selecting each option. If LEDs and the vibrator motor are not operable, contact Motorola Enterprise Mobility support (see *Service Information on page xxi*).

UCA Server System Error Code Number Index

Client Generated Error Messages

Table 14-6 below describes the five sets of client generated error messages.

Table 14-6 *Client Generated Error Messages*

Error Set	Error Message	Description
Phone Related Errors (continued on next page)	"Destination timeout"	A timeout occurred waiting for the destination to answer the phone
	"Invalid call handle'	The handle used to access the phone subsystem component is invalid. Contact help desk or technical support for assistance.
	"Invalid state"	The phone call operation is not appropriate for the current state of the phone subsystem. The phone subsystem may still be functional depending on the operation attempted. Report problem to help desk.
	"Invalid init string"	The initialization string for the phone subsystem is invalid. Check user profile and UCA server setup. Report problem to help desk or technical support.
	"Destination invalid"	The destination phone number or extension is invalid when placing a call or transferring a call. Check extensions on the UCA server. If problem persists, report problem to help desk or technical support.
	"Not initialized"	An attempt was made to perform a phone subsystem operation without it being properly initialized. Check UCA server settings for phone. If problem persists report problem to help desk or technical support.
	"Not registered"	The phone subsystem is not registered to a PBX server or for direct CA50-to-CA50 IP based call functionality. Check UCA server and PBX Server settings. If problem persists, report problem to help desk or to technical support.
	"No call exists"	This error may occur if an attempt to transfer a call when one does not exist. Report problem if it is certain that a phone call exists.
	"Already registered"	This error may occur if an attempt is made to register the phone subsystem while the phone subsystem is already registered. Report problem to help desk or technical support.
	"Invalid parameter"	This error indicates that an invalid parameter was used when attempting to access the phone subsystem. Report problem to help desk or technical support.
	"Unknown error"	Report problem to help desk or technical support.

Table 14-6 *Client Generated Error Messages (Continued)*

Error Set	Error Message	Description
Phone Related Errors (continued)	"Not Implemented"	This error indicates that an attempt was made to perform an operation with the phone subsystem but the operation is not implemented. Report problem to the help desk or technical support. Report problem to help desk or technical support.
	"Invalid DTMF tone"	A DTMF tone was attempted be sent over the established phone connection but the DTMF tone was invalid. Check the user profile. If the problem still persists, report problem to help desk or technical support.
	"Registeration failed"	A registration to a PBX server, or to a gatekeeper, or direct CA50-to-CA50 IP based call registration failed. Check PBX, gatekeeper, user profile and UCA server settings. If problem still persists, report problem to help desk or technical support.
	"UnRegistration failed"	A registration to a PBX server, or to a gatekeeper, or direct CA50-to-CA50 IP based call registration failed. Check PBX, gatekeeper, user profile and UCA server settings. If problem still persists, report problem to help desk or technical support.
Menu Related Errors	"Bad or no menu data from host. Try again."	An error occurred attempting to obtain a menu from the Server. The response is not recognized.
	"Menu not found x"	x = a menu that cannot be found.
Starting Up Error	"Error Starting Up"	This message displays when the user presses the left soft key *Log* while attempting to connect to the UCA server. **Note:** If the device cannot connect to the network or find the UCA server during the login attempt, an error message displays, a beep sounds and a red LED blinks. A countdown error screen appears. When the countdown is complete, an automatic retry is attempted. If the left soft key (*Log*) is pressed from the countdown screen, a *Log* screen displays with a log of the attempted network connection error. Automatic retry is disabled while in the *Log* screen. From the *Log* screen a user can press the right soft key to go *Back* to the error screen and restart automatic retry, or press right soft key to *Cancel* and put the device into sleep mode.

Table 14-6 *Client Generated Error Messages (Continued)*

Error Set	Error Message	Description
Server Communication Related Error	"Waiting for Server Response"	This message displays when a server request call was made and a programmable timeout occurred. The client is still active but it is waiting for a response from the server before continuing.
Browser Related Errors	System Problems Retrying ... Error = xxxxx System Problems Return UCA Error = xxxxx	xxxxx is either an HTTP error in the range 400 - 599, or a WININET error in the range 12000 - 12999. "Retrying ..." indicates that the error was not severe enough to give up and the client is retrying the HTTP post or get request. "Return UCA" indicates that the error is too severe and the user should return the device to the help desk. This error code is directed at the support team to try to resolve the issue: client, server, or network related.
	Waiting For Web Server Response	This message is displayed when an HTTP Get or Post request is made and a programmable timeout occurs before a response is received from the Web Server.

UCA Server System Error Code Number Ranges

General Errors

Table 14-7 *General Error Ranges*

Error Code Range	Error Type	Page
30000 - 30999	Phone Book	*14-18*
31000 - 31999	Device Management	*14-18*
32000 - 32999	XML Profile	*14-37*
33000 - 33999	User Management	*14-36*
34000 - 34999	Login/Authentication	*14-21*
35000 - 37000	DB Maintenance	*14-22*
50000 - 50999	Phone Book	*14-18*
51000 - 51999	Device Management	*14-23*
52000 - 52999	Console Status	*14-22*
53000 - 53999	Phone Extension Management	*14-23*
54000 - 54499	System Settings	*14-35*
54500 - 54999	Error Message Processing	n/a
55000 - 55999	User Authentication	*14-36*
56000 - 56999	User Management	*14-36*
57000 - 57999	XML Profile	*14-37*
58000 - 58999	Notification	*14-38*
59000 - 59999	Metrics	*14-38*
60000 - 60499	Walkie-talkie	*14-37*
60500 - 60999	Groups	*14-39*
61000 - 61499	Roles	*14-39*
61500 - 61999	Server Logging	*14-31*
62000 - 62999	Console Users	*14-32*
72500 - 72999	Console Status Data	*14-33*

UCA Database Errors

Table 14-8 *UCA Database Error Ranges*

Error Code Range	Error Type	Page
70000 - 70499	Phone Book	*14-34*
70500 - 70999	Criteria Selection	n/a
71000 - 71499	Data Provider	n/a
71500 - 71999	DB Test	n/a
72000 -72499	Device	*14-34*
73000 - 73499	Phone Extension	*14-34*
73500 - 73999	System Settings	*14-35*
74000 - 73999	DB Connection	n/a
74500 - 74499	User Authentication	*14-36*
75000 - 75499	User Management	*14-36*
75500 -75999	Walkie-talkie	*14-37*
76000 - 76499	XML Profile	*14-37*
76500 - 76999	Error Message	*14-38*
77000 - 77499	Notifications	*14-38*
77500 - 77999	Metrics	*14-38*
78000 - 78499	Logging	n/a
78500 - 78999	Groups	*14-39*
79000 - 79499	Roles	*14-39*
79500 - 79599	Console Users Data	*14-39*
89500 - 89599	Server Files	*14-40*
95000 -96000	UCA Server Maintenance	n/a

General Errors

Phone Book

Table 14-9 *Phone Book Errors*

Error Code	Description
50032	Exception occurred while getting the list of groups to be used for the phone book. Refer to the NT Event Log for more details.
50061	Exception occurred while getting the list of extensions to be used for the phone book. Refer to the NT Event Log for more details.
50101	Exception occurred while creating the ranges of extensions to be used for the phone book. Refer to the NT Event Log for more details.
50131	Exception occurred while creating a list of extensions for a given range to be used for the phone book. Refer to the NT Event Log for more details.
50134	Unsupported phone book sort option specified. See the product documentation for supported sort options.
50161	Exception occurred while getting the list of extensions for the recent call phone book. Refer to the NT Event Log for more details.
50193	Exception occurred while creating and sorting extensions into ranges for the phone book. See the NT Event log for more details.
50221	Exception occurred while updating the *Call Parked Location* in the UCA database. See the NT Event log for more details.
50223	Error. Invalid Store Number
50224	Error. Invalid User ID
50226	UpdateCallParkedLocations: Could not update *Parked Location* in UCA Server
50251	Exception occurred while creating the list of *Parked Locations* for use in the client. See the NT Event log for more details.

Device Management

Table 14-10 *Device Management Errors*

Error Code	Description
51092	Could not retrieve the required client version for this version of UCA server from the database
51093	An Exception occurred. Could not retrieve the required client version for this version of UCA server from the database
51101	A data layer exception occurred during device registration
51105	Device registration update failed during device registration.
51107	A new device could not be registered
51108	An exception occurred during device registration. Refer to the NT Event Log for more details.

Table 14-10 *Device Management Errors (Continued)*

Error Code	Description
51131	A data layer exception occurred while attempting to get the assigned store number for a given MAC address
51162	Could not retrieve the minimum client version for this version of UCA server from the database
51163	An Exception occurred. Could not retrieve the minimum client version for this version of UCA server from the database
51201	A data layer exception occurred. Device info not updated.
51205	Device update failed because of data layer error. Refer to the NT Event Log for more info.
51207	Device does not exist in the database. Update failed
51208	An exception occurred. Device update failed. Refer to the NT Event Log for more info.
51231	No devices found in the database.
51232	A data layer exception occurred while attempting to get the list of known devices. Refer to the NT Event Log for more details.
51262	A data layer exception occurred while attempting to update a device's AC status. Refer to the NT Event Log for more details.
51263	The specified device was not found in the database. AC Status update failed.
51292	A data layer exception occurred while attempting to update a device's activity by user. Refer to the NT Event Log for more details.
51293	Device not found. Update of device activity failed.
51294	Invalid location specified. Update of device activity failed.
51295	Device not associated with a user. Update of device activity failed.
51302	A data layer exception occurred while attempting to update a devices' activity by IP Address. See the NT Event log for more details.
51303	Device not found. Update of device activity failed.
51304	Invalid Store ID specified. Update of device activity failed.

XML Profile

Table 14-11 *XML Profile Errors*

Error Code	Description
32031	Exception occurred while building profile. Refer to the NT Event Log for more details.
32032	Exception occurred while releasing and removing user phone extension due to profile build failure. Refer to the NT Event Log for more details.
32033	Malformed XML in profile template. Refer to the NT Event Log for more details.
32034	Cannot build profile without user's primary group being specified. Refer to the NT Event Log for more details.

Table 14-11 *XML Profile Errors (Continued)*

Error Code	Description
32036	No matching profile template found for given user login parameters. Refer to the NT Event Log for more detail.
32061	Exception occurred while removing the user phone extension from the extension pool due to profile build failure. See the NT Eventlog for more details.
32062	Exception occurred while exporting the profile data. See the NT Eventlog for more details.
32063	No phone extensions available in the specified group. See the NT Eventlog for more details.
57034	A data layer exception occurred while attempting to get a base profile. Refer to the NT Event Log for more details.
57035	An exception occurred while attempting to get a base profile. Refer to the NT Event Log for more details.
57061	A data layer exception occurred while attempting to validate a base profile. Refer to the NT Event Log for more details.
57063	An exception occurred while validating a base profile. Error in Schema. Refer to the NT Event Log for more details.
57064	An exception occurred while validating the base profile against the schema. Refer to the NT Event Log for more details.
57065	An exception occurred while validating the base profile against the schema. Refer to the NT Event Log for more details.
57101	Profile already exists in the database.
57103	Invalid location specified.
57104	Invalid role specified. Role not defined in database.
57105	Invalid group specified. Group not defined in database.
57106	A data layer exception occurred while attempting to add a profile. Refer to the NT Event Log for more details.
57107	An exception occurred while attempting to add a profile. Refer to the NT Event Log for more details.
57134	A data layer exception occurred while attempting to get a profile. Refer to the NT Event Log for more details.
57135	An exception occurred while attempting to get a profile. Refer to the NT Event Log for more details.
57161	A similar base profile or a association exists.
57162	Attempted to update a profile that does not exist.
57164	A data layer exception occurred while attempting to update a profile. Refer to the NT Event Log for more details.
57165	An exception occurred while attempting to update a profile. Refer to the NT Event Log for more details.
57202	Attempted to remove a profile that does not exist.
57204	A data layer exception occurred while attempting to remove a profile. Refer to the NT Event Log for more details.

Table 14-11 *XML Profile Errors (Continued)*

Error Code	Description
57205	An exception occurred while attempting to remove a profile. Refer to the NT Event Log for more details.
57234	A data layer exception occurred while attempting to get a list of all profile associations. Refer to the NT Event Log for more details.
57235	An exception occurred attempting to get a list of all profile associations. Refer to the NT Event Log for more details.
57264	A data layer exception occurred while attempting to get a profile association ID. Refer to the NT Event Log for more details.
57265	An exception occurred while attempting to get a profile association ID. Refer to the NT Event Log for more details.
57301	Exception occurred while attempting to get a profile object.
57331	Exception occurred while attempting to get an XML object.
57361	An exception occurred while replacing menu references.
57362	Not all Menu Files referenced in project were found.
57401	An exception occurred while generating a walkie-talkie menu.
57431	An exception occurred while exporting a menu to file.
57461	An exception occurred while exporting a menu to file.
57531	An exception occurred while generating a profile template.

Login/Authentication

Table 14-12 *Login/Authentication Errors*

Error Code	Description
34031	Error obtaining the device's store number from the supplied MAC.
34033	An exception occurred during user authentication. Refer to the NT Event Log for more details.
34038	Input parameter error during user authentication.
34068	An exception occurred during user authentication. Refer to the NT Event Log for more details.
34162	Error obtaining the device's store number from the supplied MAC.
34165	Error occurred while sending an asynchronous log off command to a device.
34166	Specified user was not found in the UCA database.
34168	Error obtaining the device's I P address.
34170	An exception occurred during user authentication. Refer to the NT Event Log for more details.

DB Maintenance

Table 14-13 *DB Maintenance Errors*

Error Code	Description
35031	Database level exception generated while purging Debug Trace.
35032	Exception occurred while exporting the Debug Trace. Refer to the NT Event Log for more details.
35091	Database level exception generated while exporting Debug Trace.
35092	Exception occurred while updating DB maintenance record to reflect that the user idle check occurred. Refer to the NT Event Log for more details.
36031	Database level exception generated while performing DB Maintenance.
36032	Exception occurred while performing DB Maintenance. Refer to the NT Event Log for more details.
36061	Database level exception generated while exporting Debug Trace.
36062	Exception occurred while exporting and/ purging the Debug Trace. Refer to the NT Event Log for more details.
36091	Database level exception generated while configuring DB Maintenance.
36092	Exception occurred while configuring DB Maintenance. Refer to the NT Event Log for more details.

Console Status

Table 14-14 *Console Status Errors*

Error Code	Description
52031	No rows were found for the specified lookup
52032	A data layer exception occurred while attempting to get a database lookup table
52062	A data layer exception occurred while attempting to get the system status lookup table
52092	A data layer exception occurred while attempting to get add a users status
52102	A data layer exception occurred while attempting to add a device's status
52132	A data layer exception occurred while attempting to add database maintenance status
52162	A data layer exception occurred while attempting to add the server versions info
52191	A data layer exception occurred while attempting to add misc status

Phone Extension Management

Table 14-15 *Phone Extension Management Errors*

Error Code	Description
53031	The specified phone extension already exists in the database.
53032	SCCP MAC cannot be NULL.
53034	A data layer exception occurred while attempting to add a phone extension. Refer to the NT Event Log for more details.
53035	An exception occurred while attempting to add a phone extension. Refer to the NT Event Log for more details.
53037	SCCP MAC already exists.
53064	A data layer exception occurred while attempting to get information about a phone extension. Refer to the NT Event Log for more details.
53065	An exception occurred while attempting to get information about a phone extension. Refer to the NT Event Log for more details.
53091	Phone extension update failed. It does not exist in the database
53094	A data layer exception occurred while attempting to update a phone extension. Refer to the NT Event Log for more details.
53095	An exception occurred while attempting to update a phone extension. Refer to the NT Event Log for more details.
53097	SCCP MAC already exists
53102	Remove Extension Failed. The extension does not exist in the database.
53104	A data layer exception occurred while attempting to remove a phone extension. Refer to the NT Event Log for more details.
53105	An exception occurred while attempting to remove a phone extension. Refer to the NT Event Log for more details.
53131	No phone extensions available for assignment.
53133	Phone extension assignment to a user failed
53134	Phone extension was assigned to a user.
53135	An exception occurred while attempting to assign a phone extension. Refer to the NT Event Log for more details.
53136	A data layer exception occurred while attempting to assign a phone extension. Refer to the NT Event Log for more details.
53161	Update of extension assignment failed
53164	A data layer exception occurred while attempting to update a phone extension assignment. Refer to the NT Event Log for more details.

Table 14-15 *Phone Extension Management Errors (Continued)*

Error Code	Description
53165	An exception occurred while attempting to update a phone extension assignment. Refer to the NT Event Log for more details.
53191	An error occurred while attempting to release a phone extension.
53194	A data layer exception occurred while attempting to release a phone extension. Refer to the NT Event Log for more details.
53195	An exception occurred while attempting to release a phone extension. Refer to the NT Event Log for more details.
53202	Could not update the user's phone extension.
53203	Could not get an extension for assignment during request for a new one.
53204	Could not mark bad extension during request for a new one.
53205	An exception occurred while attempting to request a new phone extension. Refer to the NT Event Log for more details.
53206	A data layer exception occurred while attempting to request a new phone extension. Refer to the NT Event Log for more details.
53231	No non UCA extensions defined in the database.
53234	A data layer exception occurred while attempting to get a list of non UCA phone extensions. Refer to the NT Event Log for more details.
53235	An exception occurred while attempting to get a list of non UCA phone extensions. Refer to the NT Event Log for more details.
53261	No UCA extensions defined in the database.
53264	A data layer exception occurred while attempting to get a list of UCA phone extensions. Refer to the NT Event Log for more details.
53265	A data layer exception occurred while attempting to get a list of UCA phone extensions. Refer to the NT Event Log for more details.
53291	No recently assigned phone extensions in the database.
53294	A data layer exception occurred while attempting to get a list of recently assigned UCA phone extensions. Refer to the NT Event Log for more details.
53295	An exception occurred while attempting to get a list of recently assigned UCA phone extensions. Refer to the NT Event Log for more details.
53301	Error occurred while parsing a phone extension CSV file.
53304	A data layer exception occurred while attempting to import a phone extension CSV file. Refer to the NT Event Log for more details.
53305	An exception occurred while attempting to import a phone extension CSV file. Refer to the NT Event Log for more details.
53331	A data layer exception occurred while cleaning phone extension assignment Records. Refer to the NT Event Log for more details.

Table 14-15 *Phone Extension Management Errors (Continued)*

Error Code	Description
53332	An exception occurred while cleaning phone extension assignment Records. Refer to the NT Event Log for more details.
53333	Error occurred while getting the list of all phone extension assignments during phone extension maintenance. Refer to the NT Event Log for more details.
53334	Error occurred while removing an extension assignment during phone extension maintenance. Refer to the NT Event Log for more details.
53335	Error occurred while cleaning a user extension assignment during phone extension maintenance. Refer to the NT Event Log for more details.
53361	SCCP Mac already exists, it must be unique.
53364	A data layer exception occurred while attempting to get the SCCP MAC. Refer to the NT Event log for more details.
53365	An exception occurred while attempting to Get the SCCP MAC Info. Refer to the NT Event log for more details.
53394	A data layer exception occurred while attempting to Set a Phone Extension Invalid. Refer to the NT Event log for more details.
5339	An exception occurred while attempting to Set a Phone Extension Invalid. Refer to the NT Event log for more details.
53411	A data layer exception occurred while attempting to get the PetID of the User. Refer to the NT Event log for more details.
53412	An exception occurred while attempting to get the PetID of the User. Refer to the NT Event log for more details.
53421	Update of extension status failed.
53421	Update of extension status failed.
53424	A data layer exception occurred while attempting to update a phone extension status. Refer to the NT Event log for more details.
53425	An exception occurred while attempting to update a phone extension status. Refer to the NT Event log for more details.
53431	An error occurred while attempting to release a phone extension by ownerID.
53434	A data layer exception occurred while attempting to release a phone extension by owner ID. Refer to the NT Event log for more details.
53435	An exception occurred while attempting to release a phone extension by Owner ID. Refer to the NT Event log for more details.

User Authentication

Table 14-16 *User Authentication Errors*

Error Code	Description
55031	A data layer exception occurred during user authentication
55033	The specified user was not found in the UCA database.
55034	The user is a UCA user but does not have an active status.
55061	The specified user was not found in the UCA database.
55066	A data layer exception occurred while getting user data Refer to the NT Event Log for more details.
55067	An exception occurred while getting user data. Refer to the NT Event Log for more details.
55091	The specified user is already registered as a CA50 device user.
55094	No group specified for the user. A user must be belong to at least one group.
55096	A data layer exception occurred while adding a user Refer to the NT Event Log for more details.
55097	An exception occurred while adding a user Refer to the NT Event Log for more details.
55132	An exception occurred while removing a user Refer to the NT Event Log for more details.
55133	A data layer exception occurred while removing a user Refer to the NT Event Log for more details.
55134	Cannot remove a logged on user.
55135	The specified user was not found in the UCA database.
55136	Could not remove the user from the UCA database.
55137	Could not remove the user from the UCA database.
55138	Could not remove the user's assigned group.
55140	Error occurred while getting a user's logged on status.
55162	An exception occurred while updating a user Refer to the NT Event Log for more details.
55163	A data layer exception occurred while updating a user Refer to the NT Event Log for more details.
55164	Group not specified for the user.
55165	The specified user was not found in the UCA database.
55166	User update failed.
55167	Could not update the users associated group information.
55191	Error occurred while attempting to get information about the user.
55192	An exception occurred while authenticating a user. Refer to the NT Event Log for more details.
55194	A data layer error occurred while attempting to get information about the user.
55195	An error occurred while reading the login system variable.
55196	A data layer exception occurred during user authentication.

Table 14-16 *User Authentication Errors (Continued)*

Error Code	Description
55231	An error occurred while attempting to get information about the user.
55232	An exception occurred during the login process. Refer to the NT Event Log for more details.
55234	An error occurred while reading the login system variable.
55261	A data layer exception occurred during user authentication with password.
55263	The PIN supplied is invalid.
55264	The specified user was not found in the UCA database.
55265	An exception occurred while attempting to validate the PIN. Refer to the NT Event Log for more details.
55291	No users found in the UCA database.
55294	A data layer exception occurred while getting the list of users Refer to the NT Event Log for more details.
55295	An exception occurred while getting the list of users Refer to the NT Event Log for more details.
55332	An exception occurred during device authentication. Refer to the NT Event Log for more details.
55334	A data layer exception occurred during user authentication with password.
55362	The specified user was not found and the PIN was not updated.
55363	A data layer exception occurred while updating a user PIN. Refer to the NT Event Log for more details.
55364	An exception occurred while updating a user's PIN Refer to the NT Event Log for more details.
55393	A data layer exception occurred while importing a user CSV file. Refer to the NT Event log for more details.
55394	An exception occurred while importing a user CSV file. Refer to the NT Event log for more details.
55395	User data format error. The user data in the file is not formatted properly.

Notification

Table 14-17 *Notification Errors*

Error Code	Description
58035	A data layer exception occurred while getting routing info for a asynchronous message. Refer to the NT Event Log for more details.
58036	An exception occurred while getting routing info for a asynchronous message. Refer to the NT Event Log for more details.
58065	A data layer exception occurred while getting parameter info for an asynchronous message. Refer to the NT Event Log for more details.
58066	An exception occurred while getting parameter info for an asynchronous message. Refer to the NT Event Log for more details.
58137	A data layer exception occurred while sending an asynchronous message. Refer to the NT Event Log for more details.

Table 14-17 *Notification Errors (Continued)*

Error Code	Description
58138	An exception occurred while sending an asynchronous message. Refer to the NT Event Log for more details.
58163	An Argument exception occurred while processing notification fields for an asynchronous message. Refer to the NT Event Log for more details.
58164	An exception occurred while processing notification fields for an asynchronous message. Refer to the NT Event Log for more details.
58193	An Argument exception occurred while getting a user ip address destination for an asynchronous message. Refer to the NT Event log for more details.
58194	An exception occurred while getting a user ip address destination for an asynchronous message. Refer to the NT Event log for more details.
58233	An Argument exception occurred while getting an IP address destination for an asynchronous message. Refer to the NT Event Log for more details.
58234	An exception occurred while getting an IP address destination for an asynchronous message. Refer to the NT Event Log for more details.
58261	Destination type must be walkie-talkie channel if server side audio file is specified as part of an asynchronous message.
58262	Parameter Error: Unable to process Destination Information while sending an asynchronous message. See the NT Event log for more details.
58263	Unable to retrieve routing information for sending an asynchronous message. Refer to the NT Event Log for more details.
58264	Unable to build payload for sending an asynchronous message. Refer to the NT Event Log for more details.
58265	A data layer exception occurred while sending an asynchronous message. Refer to the NT Event Log for more details.
58266	An exception occurred while sending an asynchronous message. Refer to the NT Event Log for more details.
58291	Unable to retrieve predefined asynchronous message. Refer to the NT Event Log for more details.
58292	Destination type must be walkie-talkie channel if server side audio file is specified as part of an asynchronous message.
58293	Parameter Error: Unable to process destination information while sending an asynchronous message. Refer to the NT Event Log for more details.
58294	Unable to perform string replacements for predefined asynchronous message while sending an asynchronous message.
58295	A data layer exception occurred while sending an asynchronous message. Refer to the NT Event Log for more details.
58296	An exception occurred while sending an asynchronous message. Refer to the NT Event Log for more details.
58365	A data layer exception occurred while getting parameter info for an asynchronous message. Refer to the NT Event Log for more details.

Table 14-17 *Notification Errors (Continued)*

Error Code	Description
58366	An exception occurred while getting parameter info for an asynchronous message. Refer to the NT Event Log for more details.
58391	Unsupported route Type for Sending asynchronous message. Refer to the NT Event Log for more details.
58397	A data layer exception occurred while sending an asynchronous message. Refer to the NT Event log for more details.
58398	An exception occurred while sending an asynchronous message. Refer to the NT Event log for more details.

Metrics

Table 14-18 *Metrics Errors*

Error Code	Description
59032	Error occurred while logging an incoming call
59033	Error identifying the user during logging of incoming call.
59034	A data layer exception occurred while logging an incoming call. Refer to the NT Event Log for more details.
59035	An exception occurred while logging an incoming call Refer to the NT Event Log for more details.
59062	Error occurred while logging an out going call
59063	Error identifying the user during logging of an out going call.
59064	A data layer exception occurred while logging an out going call. Refer to the NT Event Log for more details.
59065	An exception occurred while logging an out going call. Refer to the NT Event Log for more details.
59093	Error occurred while logging metrics
59094	A data layer exception occurred while logging device metrics. Refer to the NT Event Log for more details.
59095	An exception occurred while logging device metrics. Refer to the NT Event Log for more details.
59102	Error occurred while updating a user's DND status. Refer to the NT Event Log for more details.
59103	Error occurred while updating a users DND status. Could not add a metrics entry. Refer to the NT Event Log for more details.
59104	A data layer exception occurred while logging DND status. Refer to the NT Event Log for more details.
59105	An exception occurred while logging DND status Refer to the NT Event Log for more details.
59133	Error occurred while logging a user's login status. Could not add a metrics entry. Refer to the NT Event Log for more details.
59134	A data layer exception occurred while logging login status. Refer to the NT Event Log for more details.
59135	An exception occurred while logging login status Refer to the NT Event Log for more details.

Table 14-18 *Metrics Errors (Continued)*

Error Code	Description
59163	Error occurred while updating a user's login status. Could not add a metrics entry. Refer to the NT Event Log for more details.
59164	A data layer exception occurred while updating login status. Refer to the NT Event Log for more details.
59165	An exception occurred while updating login status. Refer to the NT Event Log for more details.
59193	Error occurred while logging a user logging out. Could not add a metrics entry. Refer to the NT Event Log for more details.
59194	A data layer exception occurred while logging a logout. Refer to the NT Event Log for more details.
59195	An exception occurred while logging a user logging out. Refer to the NT Event Log for more details.
59264	A data layer exception occurred while purging device metrics from the database. Refer to the NT Event Log for more details.
59265	An exception occurred while purging device metrics from the database Refer to the NT Event log for more details
59291	An exception occurred while exporting metrics from the database to a file. Refer to the NT Event Log for more details.
59331	An exception occurred while logging metrics and determining call info. Refer to the NT Event log for more details.

Groups

Table 14-19 *Groups Errors*

Error Code	Description
60531	The specified group already exists in the database.
60532	Database level exception generated while adding a group.
60562	Database level exception generated while getting group data.
60591	The specified group does not exist in the database.
60592	Database level exception generated while updating group data.
60602	The specified group does not exist in the database.
60603	Database level exception generated while removing group data.

Roles

Table 14-20 *Roles Errors*

Error Code	Description
61031	The specified role already exists in the database.
61034	A data layer exception occurred while attempting to add a role. Refer to the NT Event Log for more details.
61035	An exception occurred while attempting to add a role. Refer to the NT Event Log for more details.
61064	A data layer exception occurred while attempting to get role info. Refer to the NT Event Log for more details.
61065	An exception occurred while attempting to get role information. Refer to the NT Event Log for more details.
61091	The specified role does not exist in the database.
61094	A data layer exception occurred while attempting to update role info. Refer to the NT Event Log for more details.
61095	An exception occurred while attempting to update role information. Refer to the NT Event Log for more details.
61132	The specified role does not exist in the database.
61134	A data layer exception occurred while attempting to remove a role. Refer to the NT Event Log for more details.
61135	An exception occurred while attempting to remove a role. Refer to the NT Event Log for more details.
61161	There are no roles defined in the database.
61164	A data layer exception occurred while attempting to get a list of roles. Refer to the NT Event Log for more details.
61165	An exception occurred while attempting to get a list of roles. Refer to the NT Event Log for more details.
61192	No permissions defined for the specified role.
61194	A data layer exception occurred while attempting to get a list of permissions for a given role. Refer to the NT Event log for more details.
61195	An exception occurred while attempting to get a list of permissions for a given role. Refer to the NT Event log for more details.

Server Logging

Table 14-21 *Server Logging Errors*

Error Code	Description
61532	Exception occurred while exporting the Debug Trace. Refer to the NT Event log for more details.
61533	Exception occurred while exporting the Debug Trace. Refer to the NT Event log for more details

Table 14-21 *Server Logging Errors (Continued)*

Error Code	Description
61561	Database level exception generated while purging Debug Trace.
61591	Database level exception generated while purging the Debug Trace.
61592	Exception occurred while purging the Debug Trace. Refer to the NT Event Log for more details.

Console Users

Table 14-22 *Console Users Errors*

Error Code	Description
62031	A data layer exception occurred while attempting console authentication
62034	Authentication of the specified user and password failed
62035	The specified user does not have active status which is required for log in.
62036	Authentication of the specified user and password failed
62037	Exception occurred while attempting console authentication. Refer to the NT Event Log for more details.
62061	A data layer exception occurred while attempting console login
62062	Exception occurred while attempting console login. Refer to the NT Event Log for more details.
62063	Console login failed because of authentication error
62064	Console login failed because of user update error.
62091	No console users found in the database.
62092	A data layer exception occurred while getting the list of console users
62131	A data layer exception occurred while getting a console user's information
62161	An exception occurred while attempting to get a console user's info. Refer to the NT Event Log for more details.
62191	A data layer exception occurred while adding a console user.
62192	The specified console user is already in the database
62193	Null input data supplied to the add console user method.
62231	An exception occurred while attempting to add a console user. Refer to the NT Event Log for more details.
62261	A data layer exception occurred while removing a console user.
62262	The specified console user was not found in the database
62291	A data layer exception occurred while updating a console user.
62292	The specified user was not found in the database
62293	Null input data supplied to the update console user method.

Table 14-22 *Console Users Errors (Continued)*

Error Code	Description
62331	An exception occurred while attempting to update console user info. Refer to the NT Event log for more details.
62361	A data layer exception occurred while setting a console user's pin.
62362	The specified user was not found in the database.
62363	Database error occurred while attempting to set a console user's pin.
62391	Import of users to CSV failed.
62392	A data layer exception occurred while attempting to import user data to CSV format.
62393	An exception occurred while attempting to import user data to CSV format. Refer to the NT Event Log for more details.

Console Status Data

Table 14-23 *Console Status Data Errors*

Error Code	Description
72511	Exception occurred while checking if DB maintenance is due. Refer to the NT Event Log for more details.
72521	Exception occurred while getting last DB maintenance timestamp. Refer to the NT Event Log for more details.
72531	Exception occurred while adding last DB maintenance record. Refer to the NT Event Log for more details.
72541	Exception occurred while updating last DB maintenance. Refer to the NT Event Log for more details.
72551	Exception occurred while removing record of last DB maintenance. Refer to the NT Event Log for more details.
72571	Exception occurred while getting all DB maintenance records. Refer to the NT Event Log for more details.

UCA Database Errors

Phone Book

Table 14-24 *Phone Book Errors*

Error Code	Description
70011	Exception occurred while getting PB Group List. Refer to the NT Event Log for more details.
70021	Exception occurred while getting User Extension List By Group. Refer to the NT Event Log for more details.
70031	Exception occurred while getting UCA Ext Group List. Refer to the NT Event Log for more details.

Device

Table 14-25 *Device Errors*

Error Code	Description
72011	Exception occurred while adding device Registration. Refer to the NT Event Log for more details.
72021	Exception occurred while getting device Registration. Refer to the NT Event Log for more details.
72031	Exception occurred while updating device Registration. Refer to the NT Event Log for more details.
72041	Exception occurred while removing device Registration. Refer to the NT Event Log for more details.
72061	Exception occurred while getting all inactive devices. Refer to the NT Event Log for more details.
72071	Exception occurred while getting all devices. Refer to the NT Event Log for more details.
72081	Exception occurred while updating device AC status. Refer to the NT Event Log for more details.
72091	Exception occurred while updating device activity by user. Refer to the NT Event Log for more details.
72101	Exception occurred while updating device activity by IP address. Refer to the NT Event Log for more details.

Phone Extension

Table 14-26 *Phone Extension Errors*

Error Code	Description
73011	Exception occurred while adding phone extension. Refer to the NT Event Log for more details.
73021	Exception occurred while getting phone extension info. Refer to the NT Event Log for more details.
73031	Exception occurred while updating phone extension info. Refer to the NT Event Log for more details.
73041	Exception occurred while removing phone extension. Refer to the NT Event Log for more details.
73051	Exception occurred while getting a list of available extensions for a device user. Refer to the NT Event Log for more details.
73061	Exception occurred while assigning an extension to a specific user. Refer to the NT Event Log for more details.
73071	Exception occurred while getting an extension assignment. Refer to the NT Event Log for more details.

Table 14-26 *Phone Extension Errors*

Error Code	Description
73081	Exception occurred while removing an extension assignment. Refer to the NT Event Log for more details.
73091	Exception occurred while updating an extension assignment. Refer to the NT Event Log for more details.
73101	Exception occurred while getting the static extension list for phone book. Refer to the NT Event Log for more details.
73111	Exception occurred while getting all static extensions in a given range for the phone book. Refer to the NT Event Log for more details.
73121	Exception occurred while getting all non-UCA extensions. Refer to the NT Event Log for more details.
73131	Exception occurred while getting all UCA extensions. Refer to the NT Event Log for more details.
73141	Exception occurred while getting all recently assigned extensions. Refer to the NT Event Log for more details.
73181	Exception occurred while getting the PetID of the User. Refer to the NT Event log for more details.

System Settings

Table 14-27 *System Settings Errors*

Error Code	Description
73511	Exception occurred while getting the value for a system setting. Refer to the NT Event Log for more details.
73512	Specified system setting not found in DB. Refer to the NT Event Log for more details.
73521	Exception occurred while getting the boolean value for a system setting. Refer to the NT Event Log for more details.
73522	Specified system setting not found in DB. Refer to the NT Event Log for more details.
73531	Exception occurred while getting the list of tokens for all system settings. Refer to the NT Event Log for more details.
73541	Exception occurred while getting user preference token for system setting. Refer to the NT Event Log for more details.
73551	Exception occurred while getting all system settings. Refer to the NT Event Log for more details.
73561	Exception occurred while setting the value for a system setting. Refer to the NT Event Log for more details.
73571	Exception occurred while getting a system setting. Refer to the NT Event Log for more details.
73581	Exception occurred while getting list of all values for a given domain. Refer to the NT Event Log for more details.

User Authentication

Table 14-28 *User Authentication Errors*

Error Code	Description
74521	Exception occurred while adding a device user. Refer to the NT Event Log for more details.
74531	Exception occurred while getting a device user. Refer to the NT Event Log for more details.
74541	Exception occurred while removing a device user. Refer to the NT Event Log for more details.
74551	Exception occurred while updating a device user. Refer to the NT Event Log for more details.
74561	Exception occurred while adding a group to a device user. Refer to the NT Event Log for more details.
74571	Exception occurred while getting a specific device user group association. Refer to the NT Event Log for more details.
74581	Exception occurred while removing a specific device user group association. Refer to the NT Event Log for more details.
74591	Exception occurred while updating a specific device user group association. Refer to the NT Event Log for more details.
74601	Exception occurred while getting all groups for a device user. Refer to the NT Event Log for more details.
74611	Exception occurred while setting a device user's PIN. Refer to the NT Event Log for more details.

User Management

Table 14-29 *User Management Errors*

Error Code	Description
75011	Exception occurred while adding a device user to the cache. Refer to the NT Event Log for more details.
75021	Exception occurred while getting a device user from the cache. Refer to the NT Event Log for more details.
75031	Exception occurred while removing a device user from the cache. Refer to the NT Event Log for more details.
75041	Exception occurred while updating a device user in the cache. Refer to the NT Event Log for more details.
75051	Exception occurred while updating a device user login time in the cache. Refer to the NT Event Log for more details.
75061	Exception occurred while adding device user group in cache. Refer to the NT Event Log for more details.
75071	Exception occurred while getting all groups for a device user from cache. Refer to the NT Event Log for more details.
75081	Exception occurred while removing all groups for a device user from cache. Refer to the NT Event Log for more details.
75101	Exception occurred while updating a device user *Do not disturb* status in the cache. Refer to the NT Event Log for more details.

Table 14-29 *User Management Errors (Continued)*

Error Code	Description
75111	Exception occurred while updating a device user's activity timestamp in the cache. Refer to the NT Event Log for more details.
75121	Exception occurred while getting device user ID from cache by IP address. Refer to the NT Event Log for more details.
75131	Exception occurred while getting All Console Users. Refer to the NT Event Log for more details.
75141	Exception occurred while getting device user ID from cache by device ID. Refer to the NT Event Log for more details.
75151	Exception occurred while getting a device user's IP address from cache. Refer to the NT Event Log for more details.
75161	Exception occurred while adding a device user's preference to the cache. Refer to the NT Event Log for more details.
75171	Exception occurred while getting a device user's preference from the cache. Refer to the NT Event Log for more details.
75181	Exception occurred while removing a device user's preference from the cache. Refer to the NT Event Log for more details.
75191	Exception occurred while updating a device user's preference in the cache. Refer to the NT Event Log for more details.
75211	Exception occurred while getting All Console Users. Refer to the NT Event Log for more details.
75221	Exception occurred while getting a device user's preference from the cache by the system variable token. Refer to the NT Event Log for more details.
75241	Exception occurred while getting device user ID from cache by phone extension. Refer to the NT Event Log for more details.

Walkie-Talkie

Table 14-30 *Walkie-Talkie Errors*

Error Code	Description
75511	Exception occurred while adding a walkie-talkie channel. Refer to the NT Event Log for more details.

XML Profile

Table 14-31 *XML Profile Errors*

Error Code	Description
76011	Exception occurred while getting profile association for specified Role
76021	Exception occurred while getting All Console Users. Refer to the NT Event Log for more details.
76031	Exception occurred while getting a base profile association by its ID. Refer to the NT Event Log for more details.

Table 14-31 *XML Profile Errors (Continued)*

Error Code	Description
76041	Exception occurred while updating a base profile association by its ID. Refer to the NT Event Log for more details.
76051	Exception occurred while removing a base profile association by its ID. Refer to the NT Event Log for more details.
76061	Exception occurred while getting all base profile associations. Refer to the NT Event Log for more details.

Error Message

Table 14-32 *Error Message Errors*

Error Code	Description
76511	Exception occurred while getting system error message. Refer to the NT Event Log for more details.

Notifications

Table 14-33 *Notifications Errors*

Error Code	Description
77011	Exception occurred while getting the entire notification routing table. Refer to the NT Event Log for more details.

Metrics

Table 14-34 *Metrics Errors*

Error Code	Description
77511	Exception occurred while adding a metrics entry. Refer to the NT Event Log for more details.
77521	Exception occurred while adding a metrics notification log entry. Refer to the NT Event Log for more details.
77541	Exception occurred while adding metrics entry data. Refer to the NT Event Log for more details.
77542	Input file had no valid CSV records. Refer to the NT Event Log for more details.
77551	Exception occurred while getting all metrics entry types. Refer to the NT Event Log for more details.
77561	Exception occurred while getting all metrics entry modifiers. Refer to the NT Event Log for more details.
77611	Exception occurred while getting recent calls by user. Refer to the NT Event Log for more details.

Groups

Table 14-35 *Group Errors*

Error Code	Description
78511	Exception occurred while adding a group. Refer to the NT Event Log for more details.
78521	Exception occurred while getting a group. Refer to the NT Event Log for more details.
78531	Exception occurred while updating a group. Refer to the NT Event Log for more details.
78541	Exception occurred while removing a group. Refer to the NT Event Log for more details.

Roles

Table 14-36 *Role Errors*

Error Code	Description
79011	Exception occurred while adding a device user role. Refer to the NT Event Log for more details.
79021	Exception occurred while getting a device user role. Refer to the NT Event Log for more details.
79031	Exception occurred while updating a device user role. Refer to the NT Event Log for more details.
79041	Exception occurred while removing a device user role. Refer to the NT Event Log for more details.
79051	Exception occurred while getting all device user roles. Refer to the NT Event Log for more details.

Console Users Data

Table 14-37 *Console Users Data Errors*

Error Code	Description
79511	Exception occurred while getting All Console Users. Refer to the NT Event Log for more details.
79521	Exception occurred while getting a Console User. Refer to the NT Event Log for more details.
79531	Exception occurred while adding a console user. Refer to the NT Event Log for more details.
79541	Exception occurred while removing a Console User. Refer to the NT Event Log for more details.
79551	Exception occurred while updating a Console User. Refer to the NT Event Log for more details.
79561	Exception occurred while updating a Console User's password. Refer to the NT Event Log for more details.
79571	Exception occurred while updating a Console User's login status. Refer to the NT Event log for more details.

Server Files

Table 14-38 *Server Files Errors*

Error Code	Description
89501	Exception occurred while configuring DB Maintenance. Refer to the NT Event Log for more details.
89502	Input file specified for CSV import was not found. Refer to the NT Event Log for more details.
89503	Input file had no valid CSV records. Refer to the NT Event Log for more details.
89531	Exception occurred while parsing CSV file. Refer to the NT Event log for more details

Chapter 15 Localization

Overview

This chapter includes a description of the functionality of localized versions of the UCA client software on the CA50 device.

Localization applies only to the UCA client. The UCA server continues to display the User Interface in English. The language setting is selected by the administrator during UCA server software installation.

Supported languages include:

- English
- French
- Italian
- German
- Spanish.

UCA Server Localization

Although the UCA server continues to display the User Interface in English, data can be entered on the UCA console in one of the supported languages. Supported language data entry can include the following information:

- UCA device user information
  - First, last and common name
  - Groups
  - Roles.

When installing UCA server software, the administrator chooses the appropriate language to display on CA50 devices from the *Language ID* drop-down list.

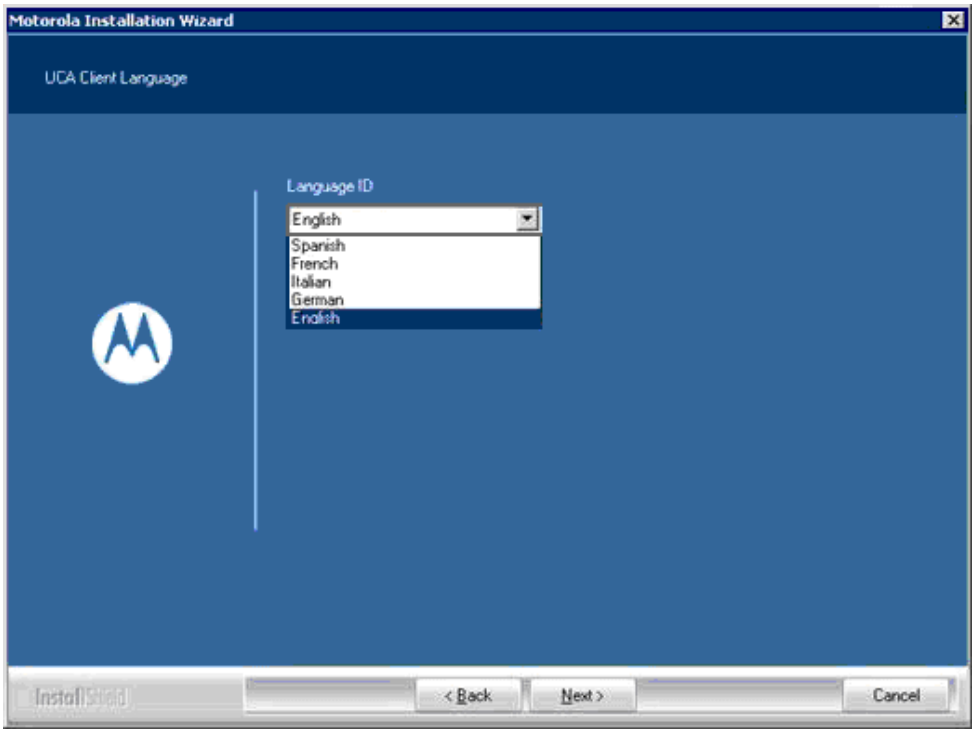


Figure 15-1 *UCA Server Installation - Language ID*

All devices associated with one UCA server use the language set on this screen.

CA50 Device Localization

No settings are required on the CA50 device to support languages other than English. However, during staging (while creating UCA settings on the MSP server using MSP 3.2.1) the appropriate language for the CA50 device must be chosen. For additional details about staging and localization see *page D-16*.

IMPORTANT When installing MSP 3.2.1, the UCA settings screen includes a *Language* drop-down list. Select the appropriate language to localize the CA50 device. The language selected must match the language selected when installing UCA server software. If the language selections do not match, the language selected during UCA server software installation is the default language on the CA50 device at login.

The CA50 device should operate in the same location as the UCA server.

NOTE All numerical data and currency symbols are in English. Currency symbols are not used in core UCA functionality. The business (Web) application developer should set up the appropriate currency formats when writing applications such as *Price Lookup*.

Business Application Localization

Localization of business applications is the responsibility of the integrator and/or developer.

The Web pages in business applications should include UTF-8 encoding to support European languages supported by the CA50 device. The following meta tag should be included in all Web pages to ensure UTF-8 encoding.

```
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
```

NOTE For sample business applications see *Appendix I, Business Applications Supplement*.

Installation

During installation of UCA server software, the User Interface prompts the administrator to select a *Language ID* from a list of supported languages for the application.Prior to installation, the Windows 2003 Operating System should be configured for the appropriate locale. The installation administrator must update the database with the appropriate changes using the install scripts to support localization.

Table 15-1 *Localization System Settings*

System Variables	Description	isLiveProfile	isUserPreference
System.Resource.Path	Path where the resource files are stored. Default Value: C:\Program files\Motorola\UCA Server\Resource Files\	False	False
System.Language.LanguageID	Locale selected during installation. Default Value: en-US	False	False

Table 15-2 *Language Culture Names*

Language Culture Name	Language
fr-FR	French (France)
it-IT	Italian
de-DE	German
es-ES	Spanish
ko-KR	Korean
ja-JP	Japanese
en-US	English

Staging

Staging is the installation step for the CA50 device. There are separate BSP images for each locale with all the necessary font and culture settings. See *Staging and Localization on page 12-6* for more information about staging.

System.Language.LanguageID, also known as the *language culture name,* is a string. *Table 15-2* lists the supported *language culture names* and their corresponding languages. Based upon the value of the *System.Language.LanguageID*, the appropriate resource file is applied and the appropriate language transmits to the CA50 device. Translations include conversions of Error messages and hard coded data sent to the device, from server.

NOTE 1.Should the system administrator enter a *Language Culture Name* not listed in *Table 15-2*, the *Language Culture Name* defaults to English (en-US).

NOTE 2.Language system variable once changed can't translate back the already existing groups or roles.

Appendix A Specifications

Technical Specifications

Table A-1 summarizes the intended operating environment and technical hardware specifications of the device.

NOTE 1. For a complete checklist outlining all the development and deployment activities required to implement a CA50 solution, see the *Development and Deployment Checklist on page 2-1*.

2. For a table listing the most commonly requested topics within this manual, see the Quick Startup Guide on the last page of the manual.

Table A-1 *CA50 Client Specifications*

Item	Description
Physical Characteristics	
Dimensions	4.37 in. L x 1.69 in. W x 1.01 in. D (111 mm x 43 mm x 25.6 mm)
Weight (includes battery)	3.75 oz./105 g
Display	Monochrome display; 96 x 64 pixel dimension
Backlight	LED backlight
Battery	Rechargeable Lithium Ion 3.7V, 920mAh
Notification	Vibrate, text, audible alert, visual LEDs
Performance Characteristics	
Operating System	**Server**: Microsoft® Windows 2003 with .NET Framework 2.0 **Symbol CA50**: Microsoft® Windows CE 5.0 Core (not accessible to developer or end-user)
Memory	64MB SDRAM/32MB Strata Flash
Interface/Communications	**Server**: Symbol MSP, RD Client, Airbeam **Symbol CA50**: Full-Speed USB 1.1 Client

Table A-1 *CA50 Client Specifications (Continued)*

Item	Description
User Environment	
Operating Temperature	14° F to 104° F / -10° C to +40° C
Storage Temperature	-40° F to 158° F / -40°C to +70°C
IP Rating	IP40
Charging Temperature	32°F to 95°F / 0° C to 35° C See *Charging Temperature on page 3-7* for detailed charging information.
Humidity	5% - 90% non-condensing
Drop Specification	3 ft. drop to concrete, 6 drops per 6 sides over operating temperature range; 4 ft. drop to concrete, 6 drops per 6 sides at ambient temperature
Electrostatic Discharge (ESD)	+/-4kVdc direct discharge, +/-8kVdc indirect discharge
Sealing	IP40
Wireless LAN Data and Voice Communications	
Antenna	Internal
Audio	Walkie-talkie: Speaker, receiver, microphone, earbud jack software support for G.729 Voice (unicast): Speaker, receiver, microphone, earbud jack: software support for G.711 u-law and G.711 A-law
Data Rates Supported	1, 2, 5.5, 6, 9, 11, 12, 18, 24, 36, 48, and 54 Mbps
Operating Channels	1-32 Walkie-talkie
Security	WPA-TKIP, Open, WEP 40, WEP 128
Voice Communication	Integrated Voice-over-IP ready (PBX, walkie-talkie), IEEE 802.11a/b/g direct sequence wireless LAN. Supports voice (unicast) calls and multicast walkie-talkie functionality.
Wireless Local Area (WLAN) radio	Conexant/USI Jedi 802.11a/b/g radio module
Data Capture Specifications	
Working Range	Typical working range on UPC 100% bar codes is 2 in. to 21 in. from the scan window.
1D Scanner (SE950) Specifications	
Roll	100% UPC at 5 in. ± 35° from vertical
Pitch Angle	Condition: 100% UPC at 5 in. ± 65° from normal

Table A-1 *CA50 Client Specifications (Continued)*

Item	Description
Skew Tolerance	Condition: 100% UPC at 5 in. ± 50° from normal
Ambient Light	Sunlight:10,000 ft. candles (107,640 lux) Artificial Light: 450 ft. candles (4,844 lux)
Scan Repetition Rate	104 (± 12) scans/sec (bidirectional)
Scan Angle	Narrow: 35 +/- 3 (typical)
Laser Power	1.7mW nominal peak power

CA50 Accessory Specifications

Table A-2 *Single-Slot Cradle Specifications*

Feature	Description
Dimensions	3.9 in. L x 3.2 in. W x 1.9 in. H (9.85 cm L x 8.23 cm W x 4.95 cm H)
Weight	4.1 oz (115 g)
Power	5.4 VDC
Typical Wall Power	6W
Storage Temperature	-40° F to 158° F / -40°C to +70°C
Charging Temperature	32°F to 95°F / 0° C to 35° C See *Charging Temperature on page 3-7* for detailed charging information.
Humidity	5% to 95% non-condensing
Drop	6 drops, 1 per side; 30.0 in. (76.2 cm) drops to vinyl tiled concrete at room temperature
Electrostatic Discharge (ESD)	+/-4kVdc direct discharge, +/-8kVdc indirect discharge

Table A-3 *Eight-Slot Charging Cradle Specifications*

Feature	Description
Dimensions	4.96 in. L x 5.47 in. W x 2.36 in. H (12.60 cm L x 13.90 cm W x 6.00 cm H)
Weight	10.1 oz
Power	12 V
Typical Wall Power	27W
Storage Temperature	-40° F to 158° F / -40°C to +70°C
Charging Temperature	32°F to 95°F / 0° C to 35° C See *Charging Temperature on page 3-7* for detailed charging information.
Humidity	5% to 95% non-condensing
Drop	6 drops, 1 per side; 30.0 in. (76.2 cm) drops to vinyl tiled concrete at room temperature
Electrostatic Discharge (ESD)	+/-4kVdc direct discharge, +/-8kVdc indirect discharge

Table A-4 *Eight-Slot Spare Battery Charger Specifications*

Feature	Description
Dimensions	4.95 in. L x 5.50 in. W x 2.36 in. H (12.56 cm L x 13.96 cm W x 5.99 cm H)
Weight	10.2 oz (286 g)
Power	12 V
Typical Wall Power	20W
Storage Temperature	-40° F to 158° F / -40°C to +70°C
Charging Temperature	32°F to 95°F / 0° C to 35° C See *Charging Temperature on page 3-7* for detailed charging information.
Humidity	5% to 95% non-condensing
Drop	6 drops, 1 per side; 30.0 in. (76.2 cm) drops to vinyl tiled concrete at room temperature
Electrostatic Discharge (ESD)	+/-4kVdc direct discharge, +/-8kVdc indirect discharge

(Page intentionally blank.)

Appendix B Accessories

Overview

CA50 accessories provide a variety of product support capabilities. Accessories include USB cable, cradles, spare battery charger, wall mounts, lanyard, soft case and earbud. See *Table B-1* for a full list of accessories and descriptions.

NOTE 1. For a complete checklist outlining all the development and deployment activities required to implement a CA50 solution, see the *Development and Deployment Checklist on page 2-1*.

2. For a table listing the most commonly requested topics within this manual, see the Quick Startup Guide on the last page of the manual.

Accessories Summary

*Table B-1*lists the accessories available for the CA50.

Table B-1 *CA50 Accessories*

Accessory	Description	Page
Cradles		
Eight-slot CA50 charger with power supply and DC line cord; (mounting bracket sold separately)	Eight-slot cradle for charging up to eight CA50 devices, simultaneously.	*B-3*
Eight-slot battery charger with power supply and DC line cord; (mounting bracket sold separately)	Eight-slot battery charger for charging up to eight CA50 batteries, simultaneously.	*B-8*
Single-slot CA50 and spare battery charger with power supply and DC line cord	Single-slot cradle for charging one CA50 (with battery installed) and one spare battery simultaneously.	*B-5*
Battery Accessories		
Eight-slot battery charger with power supply and DC line cord; (mounting bracket sold separately).	Eight-slot battery charger for charging up to eight CA50 batteries, simultaneously.	*B-8*
Spare Battery Kit	Single battery or 8 battery kits available.	n/a
Miscellaneous Accessories		
Earbud	A monaural earbud with microphone can be attached to the CA50 for telephony, VoIP or audio playback applications. Upon insertion of an earbud jack, the CA50 automatically disables its internal speaker, microphone and receiver. The CA50 is compatible with standard off the shelf earbuds.	*B-10*
Lanyard (with clip)	Used to wear the CA50 around user's neck.	*B-11*
Server software	The UCA server software uses a database to store the data for configuring, maintaining and tracking CA50s.	n/a
Soft Pouch	Optional carrying case for the CA50 (p/n 11-104787-01R).	n/a
Wall mount bracket	Available for wall mounting the multi-slot device charger and spare battery charger.	*B-13*

Eight-Slot CA50 Charging Cradle

This section describes how to set up and use an eight-slot cradle with the device.

NOTE 1. Only genuine Motorola batteries (p/n 82-83172-01) should be used with CA50 devices and charging accessories.

2. Use only an approved power supply KT-14000-148R output rated 12 Vdc and minimum 3.3 A with AC line cord 23844-00-00. The power supply is certified to EN60950-1 with SELV outputs. Use of an alternative power supply invalidates any approval given to this device and may be dangerous.

The eight-slot CA50 charging cradle:

- Provides 5.1 VDC power for operating the device.
- Charges the CA50 battery in the device.

Setup

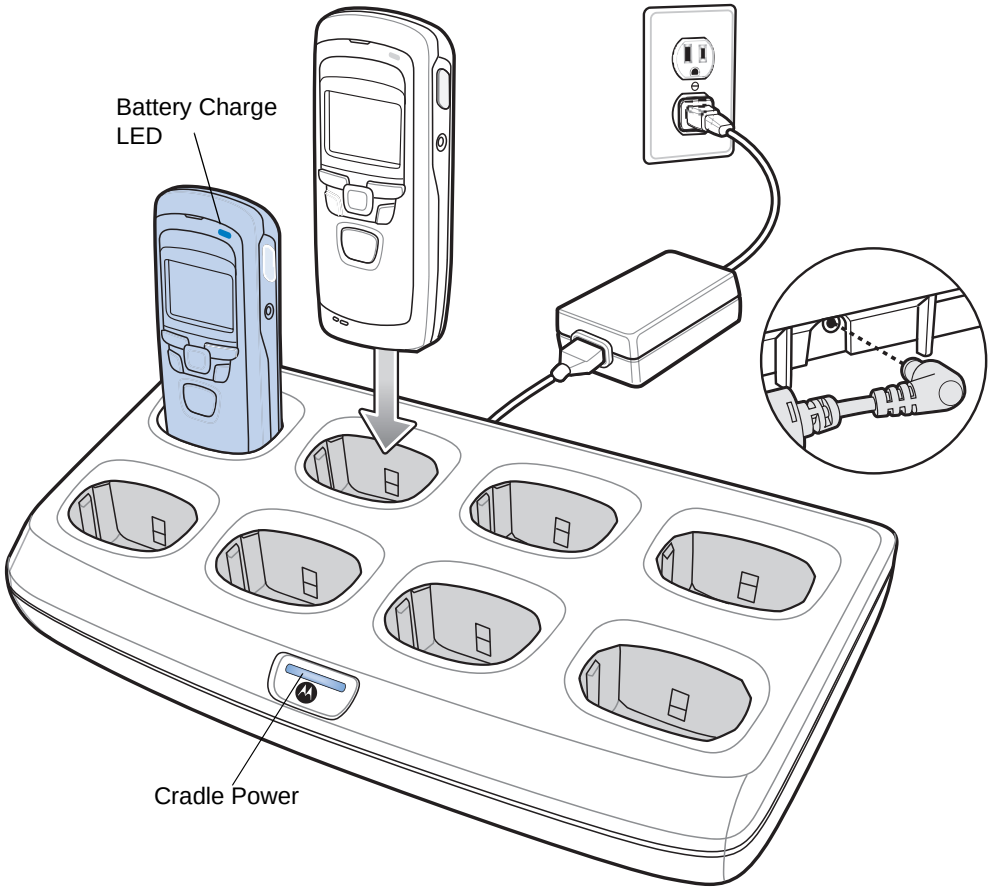


Figure B-1 *Eight-Slot Cradle Power Connection*

Battery Charging Indications

The eight-slot cradle charges up to eight CA50 devices simultaneously.

The LED on each CA50 indicates the status of the battery charging in the device. See *Table 4-3 on page 4-10* for charging status indications.

Batteries nominally take three hours to charge at ambient temperatures of 30° C or below.

Charging Temperature

See *Charging Temperature on page 3-7* for detailed information.

RF Connectivity

When a CA50 is inserted into a charging cradle, it maintains RF communication over the wireless network.

NOTE CA50 cradles do not support wired ethernet communication.

Single-Slot CA50 Charging Cradle with Spare Battery Charger

This section describes how to set up and use a single-slot cradle with the device.

NOTE 1. Only genuine Motorola batteries (p/n 82-83172-01) should be used with CA50 devices and charging accessories.

2. Use only an approved power supply 50-14000-147R output rated 12 Vdc and minimum 3.3 A with AC line cord 50-16000-182R. The power supply is certified to EN60950-1 with SELV outputs. Use of an alternative power supply invalidates any approval given to this device and may be dangerous.

The single-slot CA50 charging cradle:

- Provides 5.1 VDC power for operating the device.
- Charges the CA50 battery in the device.
- Charges one spare battery.

Setup

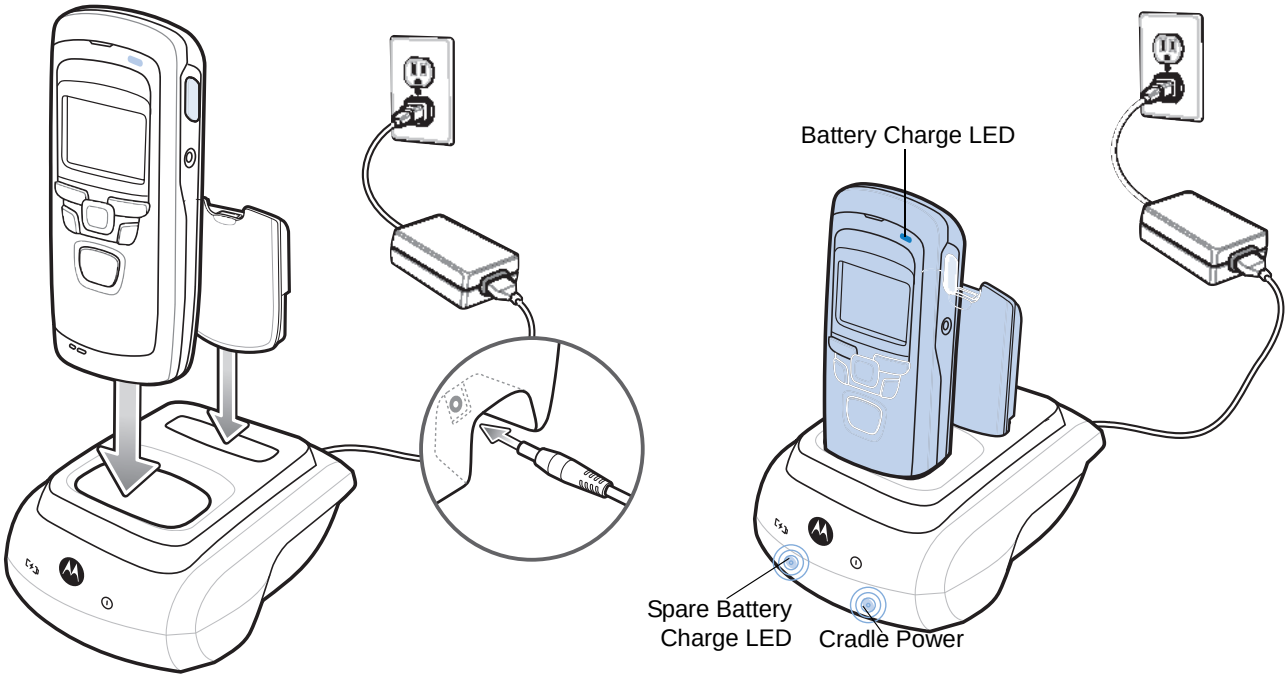


Figure B-2 *Single-Slot Cradle Power Connection*

Charging the Battery

Connect the cradle to power. Insert the CA50 into the device slot to begin charging.

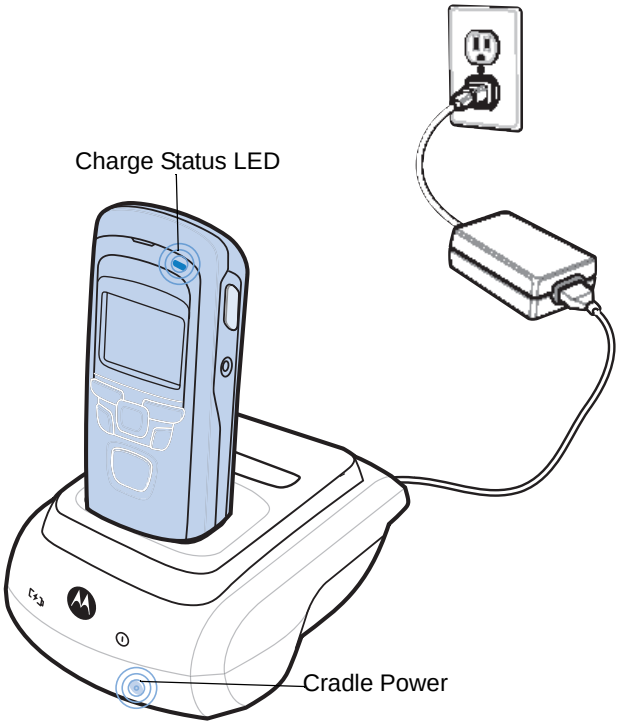


Figure B-3 *Battery Charging*

Charging the Spare Battery

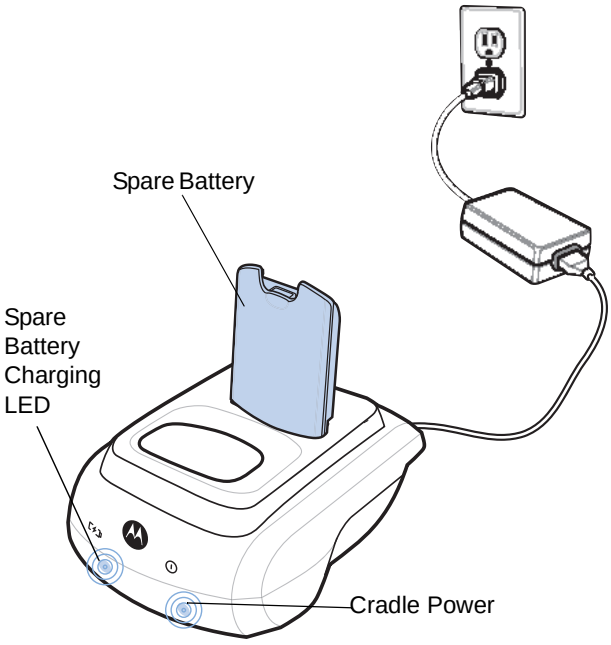


Figure B-4 *Spare Battery Charging*

Battery Charging Indications

The single- slot cradle charges the device battery and a spare battery simultaneously.

The device charge LED indicates the status of the battery charging in the device. See *Table 4-3 on page 4-10* for charging status indications.

The spare battery charging LED on the cradle indicates the status of the spare battery charging in the cradle. See *Table B-2* for charging status indications.

The battery fully charges in approximately three hours.

Spare Battery Charging Indications

Table B-2 *Spare Battery Charging Indications*

Spare Battery LED (on cradle)	Indication
Solid Red	Spare battery is charging.
Solid Green	Spare battery is fully charged.
Blinking Red	Charging error.
Off	No battery inserted.
Cradle Power	
Solid Green	Power on.
Off	Power off.

Charging Temperature

See *Charging Temperature on page 3-7* for detailed information.

RF Connectivity

When a CA50 is inserted into a charging cradle, it maintains RF communication over the wireless network.

NOTE CA50 cradles do not support wired ethernet communication.

Eight-Slot Spare Battery Charger

This section describes how to use the eight-slot spare battery charger to charge up to eight spare batteries.

NOTE 1. Only genuine Motorola batteries (p/n 82-83172-01) should be used with CA50 devices and charging accessories.

2. Use only an approved power supply KT-14000-148R output rated 12 Vdc and minimum 3.3 A with AC line cord 23844-00-00. The power supply is certified to EN60950-1 with SELV outputs. Use of an alternative power supply invalidates any approval given to this device and may be dangerous.

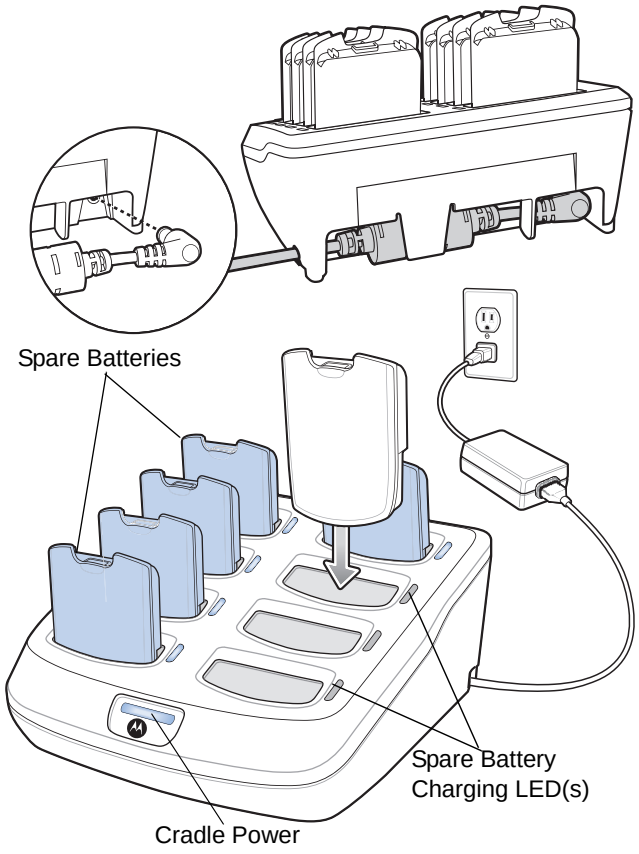


Figure B-5 *Spare Battery Installation*

Spare Battery Charging

1. Connect the charger to a power source.
2. Insert the spare battery into a spare battery charging well and gently press down on the battery to ensure proper contact.

Battery Charging Indications

An LED is provided for each battery charging well. See *Table B-3* for charging status indications. The battery fully charges in approximately three hours.

Charging Temperature

See *Charging Temperature on page 3-7* for detailed information.

LED Indications

Table B-3 *LED Indications*

LED	Indication
Solid Green	Spare battery charging is complete.
Solid Red	Spare battery is charging
Blinking Red	Charge error; check placement of spare battery.
Off	No battery inserted.

Earbud

You can use a mono earbud for audio communication when using an audio-enabled application. To use an earbud, plug the earbud (2.5mm jack) into the mono audio connector on the side of the CA50. Plugging an earbud into the jack disables the internal speaker, microphone and receiver.

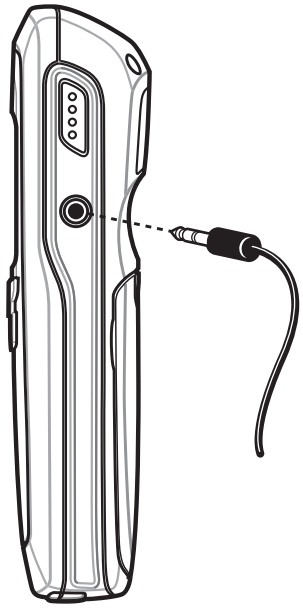


Figure B-6 *Mono Earbud Connection*

Lanyard with Clip

For ease of use, the CA50 can be attached to the lanyard with clip enabling the device to be worn around the neck as shown in *Figure B-7*.

Figure B-7 *Wearing Lanyard*

Attaching and Removing the Lanyard

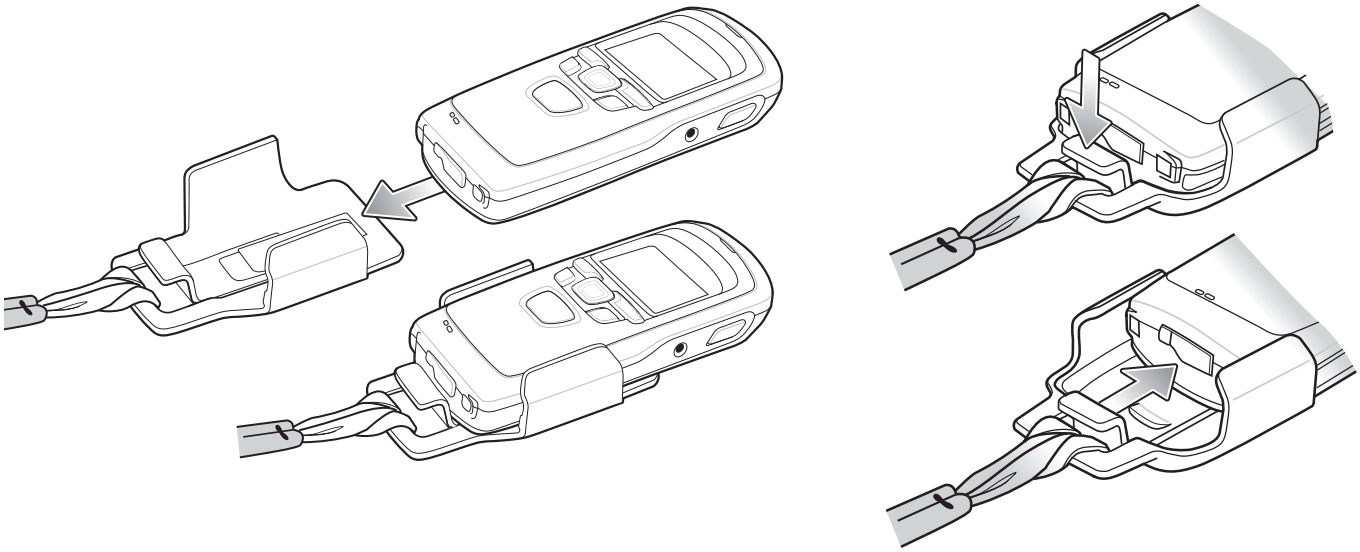


Figure B-8 *Attaching and Removing the Lanyard*

USB Cable

This section describes how to set up and use the cable. The USB cable is available to use for communication between the CA50 and a PC. It cannot be used to charge the battery in the CA50.

Communication Setup

To connect a device communication cable to a USB device:

1. Connect the USB A end of the device communication cable to the communication port of the PC.
2. Connect the USB Mini-B end of the device communication cable to the device. For more information on communication setup procedures, see *Appendix G, ActiveSync*.

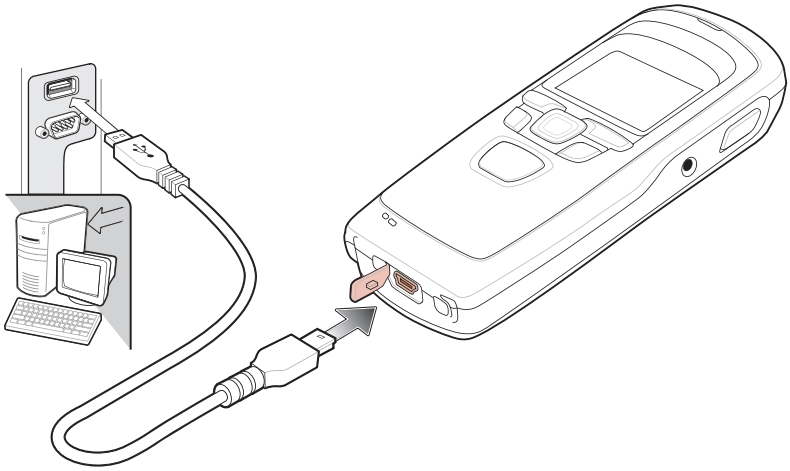


Figure B-9 *ActiveSync Cable Communication*

NOTE The CA50 cannot be charged over an ActiveSync connection.

Wall Mount Brackets

Two optional wall mount brackets are available for mounting a CA50 battery charger and/or a spare battery charger to a wall.

- The KT-102376-01R wall mount bracket is used to mount one SAC5000-8000CR (eight-slot) battery charger.
- The KT-102375-01R wall mount bracket is used to mount one CHS5000-8000CR (eight-slot) CA50 charger, or two SAC5000-8000CR (eight-slot) battery chargers.

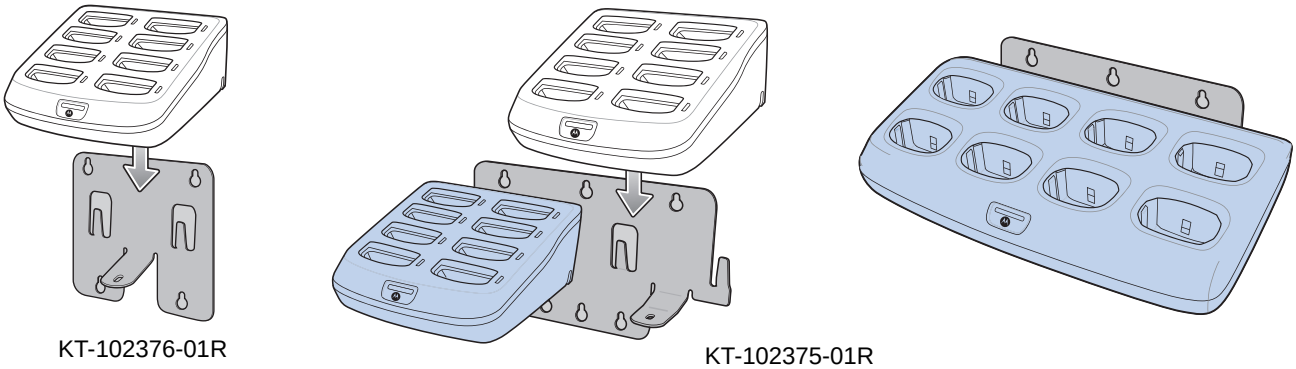


Figure B-10 *Wall Mount Brackets*

KT-102376-01R Bracket

The KT-102376-01R bracket mounts one SAC5000-8000CR (eight-slot) spare battery charger to a wall. Use the wall mount bracket as a template to mark the locations of the four mounting screws.

Included Hardware

- One KT-102376-01R Mounting Bracket
- One black .5” Phillips round head, thread cutting screw.

NOTE Use the .5” Phillips round head screw supplied in mounting bracket kit to attach the eight-slot spare battery charger to the mounting bracket. This screw is not intended for mounting the bracket on a wall.

For safety and stability, it is recommended that you use the appropriate wall mounting hardware for installation. For safe mounting, it is essential to use wall anchors appropriate to the wall type (i.e. plaster, drywall, concrete, etc.). Mount to wood studs whenever possible.

The wall mount bracket mounting slots are designed for a fastener with a #8 pan head.

Mounting Instructions

NOTE Connect an approved power supply to the cradle prior to attaching the cradle to the wall mount bracket. See *Eight-Slot Spare Battery Charger on page B-8* for power supply information.

1. Slide the two bracket mounting tabs into the receptacles on the bottom of the cradle as shown. Ensure the screw hole in the bracket flange aligns with the screw hole in the cradle.

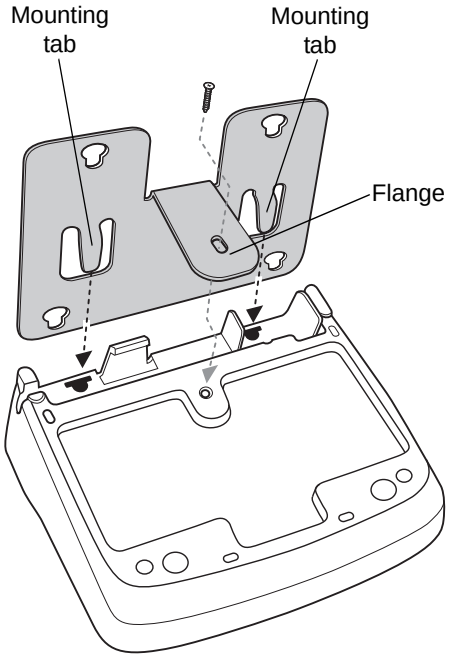


Figure B-11 *Insert Bracket into Cradle*

2. Screw the .5” Phillips screw (supplied) into the bracket flange and tighten until the cradle is securely attached to the bracket.

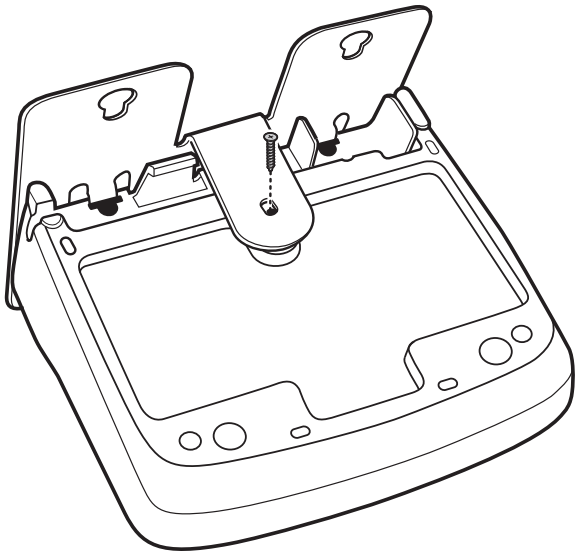


Figure B-12 *Attach Flange with Screw*

3. Use four screws to mount the bracket to a wall. Ensure to use additional wall mounting hardware, as needed, for safe mounting to the wall type.

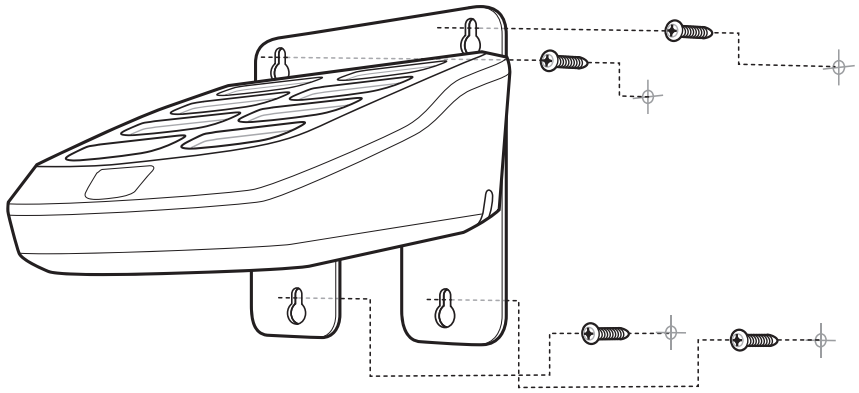


Figure B-13 *Mount to Wall*

KT-102375-01R Bracket

The KT-102375-01R bracket mounts one CHS5000-8000CR (eight-slot) CA50 charger, or two SAC5000-8000CR (eight-slot) spare battery chargers to a wall. Use the wall mount brackets as a template to mark the locations of the four mounting screws.

Included Hardware

- Two KT-102375-01R Mounting Bracket
- One black .5” Phillips round head, thread cutting screw.

NOTE Use the .5” Phillips round head screw supplied in mounting bracket kit to attach the eight-slot spare battery charger to the mounting bracket. This screw is not intended for mounting the bracket on a wall.

For safety and stability, it is recommended that you use the appropriate wall mounting hardware for installation. For safe mounting, it is essential to use wall anchors appropriate to the wall type (i.e. plaster, drywall, concrete, etc.). Mount to wood studs whenever possible.

The wall mount bracket mounting slots are designed for a fastener with a #8 pan head.

Mounting Instructions

NOTE Connect an approved power supply to the cradle(s) prior to attaching the cradle(s) to the wall mount bracket. See *Eight-Slot Spare Battery Charger on page B-8* and *Eight-Slot CA50 Charging Cradle on page B-3* for power supply information.

1. When mounting one eight-slot CA50 battery charger (CHS5000-8000CR), slide the four bracket mounting tabs into the receptacles on the bottom of the charger as shown in *Figure B-14*. Ensure the screw holes in the bracket flanges align with the screw holes in the cradle.

   When mounting two eight-slot battery chargers (SAC5000-8000CR), slide two bracket mounting tabs into each battery charger as shown in *Figure B-15*.

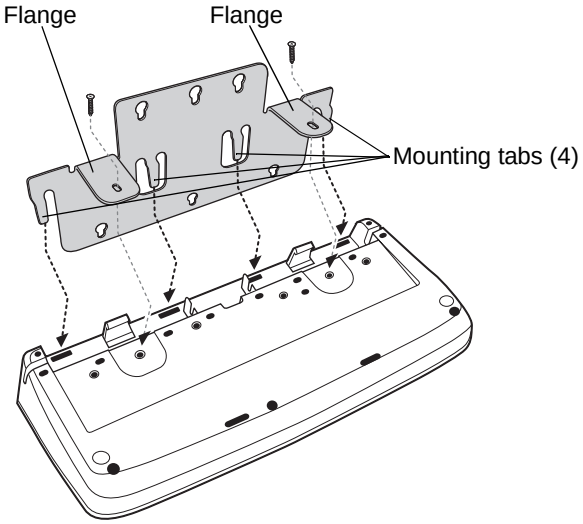


Figure B-14 *Insert Bracket into CHS5000-8000CR Cradle*

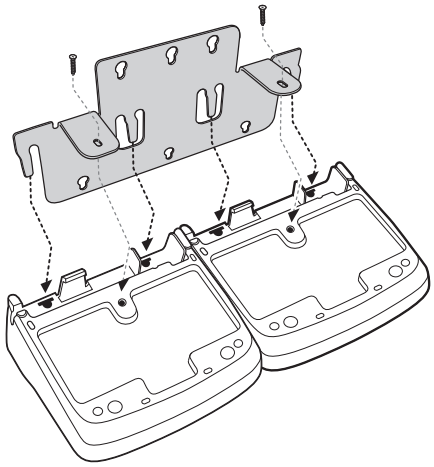


Figure B-15 *Insert Bracket into SAC5000-8000CR Cradle*

2. Screw the bracket to the cradle(s).
   CHS5000-8000CR: Screw the two supplied .5” Phillips screws into the two bracket flanges and tighten until the cradle is securely attached to the bracket (as shown in drawing A, *Figure B-16*).

   SAC5000-8000CR: Screw one supplied .5” Phillips screw in each flange and tighten one flange to each cradle (as shown in drawing B, *Figure B-16*).

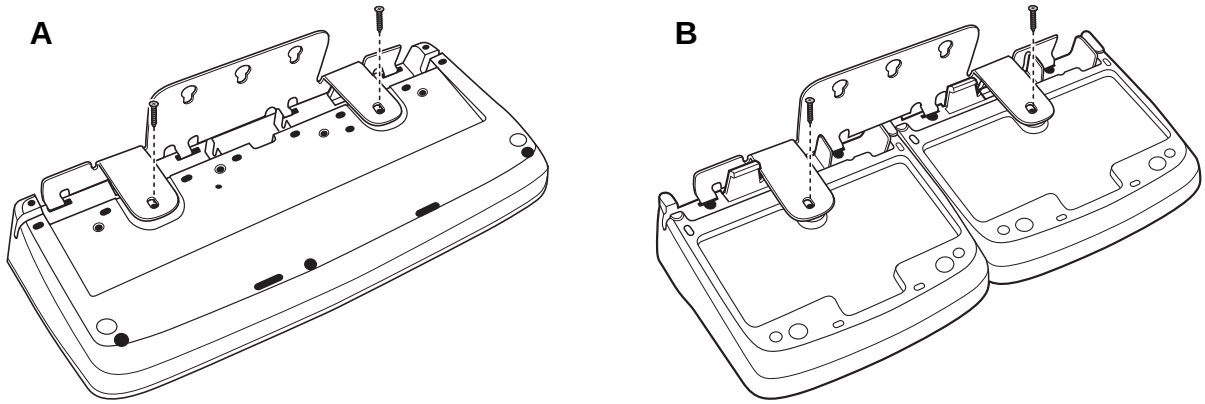


Figure B-16 *Screw the Bracket to the Cradle*

3. Use six screws (not supplied) to mount the bracket(s) to a wall.

NOTE Ensure to use additional wall mounting hardware, as needed, for safe mounting to the wall type.

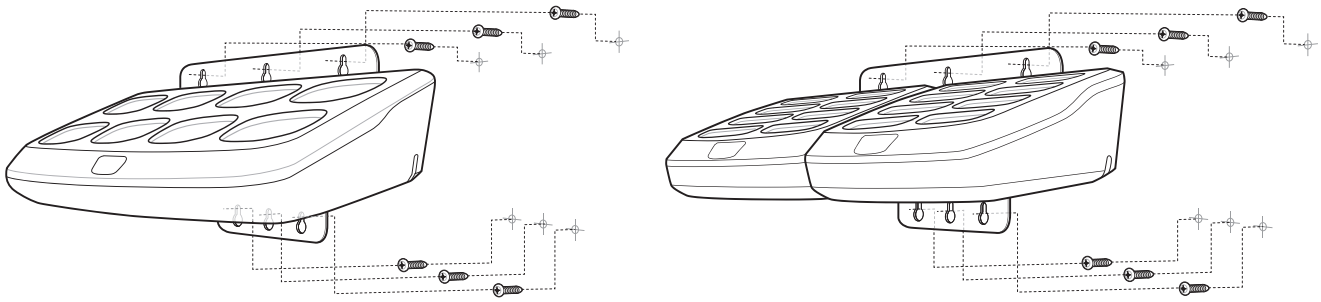


Figure B-17 *Mount Bracket(s)*

(Page intentionally blank.)

Appendix C Enabling/Disabling Scanner Symbologies

Overview

This chapter describes the 1-dimensional symbologies supported by the SE950 scan engine in the CA50, and how to enable and change the parameters of the scanner symbologies.

Enabling/Disabling Symbologies

The CA50 has several symbologies enabled by default (*page C-2*). In addition, several symbologies are supported but not enabled (*page C-2*). Both default and supported symbologies can be enabled and/or disabled by the developer.

When enabled, the parameters associated with a specific bar code symbology can also be configured. Configuration examples include, setting the minimum and maximum allowable decode length for bar code symbology such as Code128 and enabling check digits for Code 39.

Enabling and disabling symbologies is achieved through HTML meta tags. This can be accomplished during the user login process (see *page C-12*) or in Web based business applications (see *page C-12*).

Scanner configuration is implemented as proprietary HTML elements in the browser.

NOTE Chinese 2 of 5 and Decode Pointer configuration is not supported.

Supported Code Types

The CA50 supports the following code types:

- UPC-E0
- UPC-A
- UPCE1
- EAN-8
- EAN-13
- Trioptic39
- Code 128
- Code 39
- Code 93
- Interleaved 2 of 5
- Code11
- Discrete 2 of 5
- Codabar
- MSI
- Webcode
- Chinese 2 of 5
- Decode_Pointer
- GS1 DataBar-14
- GS1 DataBar Limited
- GS1 DataBar Expanded.

Enabled Code Types

The CA50 has the following code types enabled by default:

- UPC-E0
- UPC-A
- MSI
- EAN-8
- EAN-13
- Codabar
- Code 39
- Interleaved 2 of 5
- Code 128.

Configuring Symbologies

Proprietary HTML elements allow business application developers to configure symbologies based on the application requirement.

The UCA client supports proprietary HTML elements to enable the scanner, disable the scanner, and scan and read bar code data.UCA client also supports configuring the scanner symbologies and their parameters. The supported proprietary HTML elements (or custom meta–tags) follow the same convention as the proprietary HTML elements implemented in the Symbol Pocket Browser (SPB).

Table C-1 lists the custom meta tags to use to configure symbologies and their associated parameter/properties.

Table C-1 *Custom Meta Tags*

Symbology Property	Description	Custom Tags	Parameters	Example:
MSI				
MSI	The msi META Tag is an action tag used to enable or disable the msi decoder. By default enabled	<META HTTP-Equiv="msi" content="[parameter]">	Enabled - enables the decoder Disabled - disables the decoder	<META HTTP-Equiv="msi" Content="enabled"> <META HTTP-Equiv="scanner" Content="enabled">
msi-ReportCheckDigit	The msi-ReportCheckDigit META Tag is an action tag used to set the msi reportcheckdigit property.	<META HTTP-Equiv="msi-reportcheckdigit" content="[parameter]">	True - enables reporting of the check digit False - disables reporting of the check digit	<META HTTP-Equiv="msi" Content="enabled"> <META HTTP-Equiv="msi-reportcheckdigit" Content="true">
msi-MinLength META	The msi-MinLength META Tag is an action tag used to set the msi min length property.	<META HTTP-Equiv="msi-minlength" content="[parameter]">	nn 0 to 55	<META HTTP-Equiv="msi" Content="enabled"> <META HTTP-Equiv="msi-minlength" Content="30">
msi-MaxLength META	The msi-MaxLength META Tag is an action tag used to set the msi max length property.	<META HTTP-Equiv="msi-maxlength" content="[parameter]">	nn 0 to 55	<META HTTP-Equiv="msi" Content="enabled"> <META HTTP-Equiv="msi-maxlength" Content="30">
msi-CheckDigitScheme	The msi-CheckDigitScheme META Tag is an action tag used to set the msi-CheckDigitScheme property.	<META HTTP-Equiv="msi-CheckDigitScheme" content="[parameter]">	**msi_chkdgt_mod_11_10** 1st check digit is MOD 11, 2nd is MOD 10; **msi_chkdgt_mod_10_10** both check digits are MOD 10	<META HTTP-Equiv="msi-CheckDigitScheme" Content="msi_chkdgt_mod_11_10">
msi-checkdigits	The msi-checkdigits META Tag is an action tag used to set the msi checkdigits property	<META HTTP-Equiv="msi-checkdigits" content="[parameter]">	True - enables the property False - disables the property	<META HTTP-Equiv="msi" Content="enabled"> <META HTTP-Equiv="msi-checkdigits" Content="true">

Table C-1 *Custom Meta Tags (Continued)*

Symbology Property	Description	Custom Tags	Parameters	Example:
Code 128				
Code128	The Code128 META Tag is an action tag used to enable or disable the code128 decoder. By default enabled	<META HTTP-Equiv="code128" content="[parameter]">	Enabled - enables the decoder Disabled - disables the decoder	<META HTTP-Equiv="code128" Content="enabled">
Code128-Redundancy	The Code128-Redundancy META Tag is an action tag used to set the code128 redundancy property	<META HTTP-Equiv="code128-redundancy" content="[parameter]">	True - enables the property False - disables the property	<META HTTP-Equiv="code128-redundancy" Content="true">
Code128-Other128	The Code128-Other128 META Tag is an action tag used to set the code128-other128 property	<META HTTP-Equiv="code128-other128" content="[parameter]">	True - enables the other (non EAN or ISBT) 128 subtype False - disables the other (non EAN or ISBT) 128 subtype	<META HTTP-Equiv="code128-other128" Content="true">
Code128-MinLength	The Code128-MinLength META Tag is an action tag used to set the code128 min length property.	<META HTTP-Equiv="code128-minlength" content="[parameter]">	nn 0 to 55	<META HTTP-Equiv="code128-minlength" Content="30">
Code128-MaxLength	The Code128-MaxLength META Tag is an action tag used to set the code128 max length property.	<META HTTP-Equiv="code128-maxlength" content="[parameter]">	nn 0 to 55	<META HTTP-Equiv="code128-maxlength" Content="30">
Code128-ISBT128	The Code128-ISBT128 META Tag is an action tag used to set the code128-isbt128 property.	<META HTTP-Equiv="code128-isbt128" content="[parameter]">	True - enables the ISBT128 subtype False - disables the ISBT128 subtype	<META HTTP-Equiv="code128-isbt128" Content="true">
Code128-EAN128	The Code128-EAN128 META Tag is an action tag used to set the code128-ean128 property	<META HTTP-Equiv="code128-ean128" content="[parameter]">	True - enables the EAN128 subtype False - disables the EAN128 subtype	<META HTTP-Equiv="code128-ean128" Content="true">
Code 39				
Code 39	The Code39 META Tag is an action tag used to enable or disable the code39 decoder. By default enabled	<META HTTP-Equiv="code39" content="[parameter]">	Enabled - enables the decoder Disabled - disables the decoder	<META HTTP-Equiv="code39" Content="enabled">

Table C-1 *Custom Meta Tags (Continued)*

Symbology Property	Description	Custom Tags	Parameters	Example:
Code39-VerifyCheckDigit	The Code39-VerifyCheckDigit META Tag is an action tag used to set the code39 verifycheckdigit property.	<META HTTP-Equiv="code39-verifycheckdigit" content="[parameter]">	True - enables verifying of the check digit False - disables verifying of the check digit	<META HTTP-Equiv="code39-verifycheckdigit" Content="true">
Code39-ReportCheckDigit	The Code39-ReportCheckDigit META Tag is an action tag used to set the code39 reportcheckdigit property.	<META HTTP-Equiv="code39-reportcheckdigit" content="[parameter]">	True - enables reporting of the check digit False - disables reporting of the check digit	<META HTTP-Equiv="code39-reportcheckdigit" Content="true">
Code39-Redundancy	The Code39-Redundancy META Tag is an action tag used to set the code39 redundancy property	<META HTTP-Equiv="code39-redundancy" content="[parameter]">	True - enables the property False - disables the property	<META HTTP-Equiv="code39-redundancy" Content="true">
Code39-MinLength	The Code39-MinLength META Tag is an action tag used to set the code39 min length property.	<META HTTP-Equiv="code39-minlength" content="[parameter]">	nn 0 to 55	<META HTTP-Equiv="code39-minlength" Content="30">
Code39-MaxLength	The Code39-MaxLength META Tag is an action tag used to set the code128 max length property.	<META HTTP-Equiv="code39-maxlength" content="[parameter]">	nn 0 to 55	<META HTTP-Equiv="code39-maxlength" Content="30">
Code39-FullASCII	The Code39-FullASCII META Tag is an action tag used to set the code39-fullascii property.	<META HTTP-Equiv="code39-fullascii" content="[parameter]">	True - enables full ASCII conversion of the bar code False - disables full ASCII conversion of the bar code	<META HTTP-Equiv="code39-fullascii" Content="true">
Code39-ConvertToCode32	The Code39-ConvertToCode32 META Tag is an action tag used to set the code39-converttocode32 property	<META HTTP-Equiv="code39-converttocode32" content="[parameter]">	True - enables conversion of Code 39 bar codes to Code 32 False - Disables conversion of Code 39 bar codes to Code 32	<META HTTP-Equiv="code39-converttocode32" Content="true">
Code39-concatenation	The Code39-concatenation META Tag is an action tag used to set the code39-concatenation property.	<META HTTP-Equiv="code39-concatenation" content="[parameter]">	True - enables Code 39 bar code concatenation False - disables Code 39 bar code concatenation	<META HTTP-Equiv="code39-concatenation" Content="true">

Table C-1 *Custom Meta Tags (Continued)*

Symbology Property	Description	Custom Tags	Parameters	Example:
Code39-Code32Prefix	The Code39-Code32Prefix META Tag is an action tag used to set the code39-code32prefix property.	<META HTTP-Equiv="code39-code32prefix" content="[parameter]">	True - enables reporting the Code 32 prefix when a Code 39 bar code is converted False - disables reporting the Code 32 prefix when a Code 39 bar code is converted	<META HTTP-Equiv="code39-code32prefix" Content="true">
Code 93				
Code93	The Code93 META Tag is an action tag used to enable or disable the code93 decoder.	<META HTTP-Equiv="code93" content="[parameter]">	Enabled - enables the decoder Disabled - disables the decoder	<META HTTP-Equiv="code93" Content="enabled">
Code93-Redundancy	The Code93-Redundancy META Tag is an action tag used to set the code93 redundancy property	<META HTTP-Equiv="code93-redundancy" content="[parameter]">	True - enables the property False - disables the property	<META HTTP-Equiv="code93-redundancy" Content="true">
Code93-MinLength	The Code93-MinLength META Tag is an action tag used to set the code93 min length property.	<META HTTP-Equiv="code93-minlength" content="[parameter]">	nn 0 to 55	<META HTTP-Equiv="code93-minlength" Content="30">
Code93-MaxLength	The Code93-MaxLength META Tag is an action tag used to set the code93 max length property.	<META HTTP-Equiv="code93-maxlength" content="[parameter]">	nn 0 to 55	<META HTTP-Equiv="code93-maxlength" Content="30">
Interleaved 2 of 5				
i2of5	The i2of5 META Tag is an action tag used to enable or disable the i2of5 decoder. By default enabled	<META HTTP-Equiv="i2of5" content="[parameter]">	Enabled - enables the decoder Disabled - disables the decoder	<META HTTP-Equiv="i2of5" Content="enabled">
I2of5-VerifyCheckDigit	The I2of5-VerifyCheckDigit META Tag is an action tag used to set the i2of5 verifycheckdigit property.	<META HTTP-Equiv="i2of5-verifycheckdigit" content="[parameter]">	True - enables verifying of the check digit False - disables verifying of the check digit	<META HTTP-Equiv="i2of5-verifycheckdigit" Content="true">
I2of5-ReportCheckDigit	The I2of5-ReportCheckDigit META Tag is an action tag used to set the msi reportcheckdigit property.	<META HTTP-Equiv="i2of5-reportcheckdigit" content="[parameter]">	True - enables reporting of the check digit False - disables reporting of the check digit	<META HTTP-Equiv="i2of5-reportcheckdigit" Content="true">

Table C-1 *Custom Meta Tags (Continued)*

Symbology Property	Description	Custom Tags	Parameters	Example:
I2of5-Redundancy	The I2of5-Redundancy META Tag is an action tag used to set the code128 redundancy property	<META HTTP-Equiv="i2of5-redundancy" content="[parameter]">	True - enables the property False - disables the property	<META HTTP-Equiv="i2of5-redundancy" Content="true">
I2of5-MinLength	The I2of5-MinLength META Tag is an action tag used to set the code128 min length property.	<META HTTP-Equiv="i2of5-minlength" content="[parameter]">	nn 0 to 55	<META HTTP-Equiv="i2of5-minlength" Content="30">
I2of5-MaxLength	The I2of5-MaxLength META Tag is an action tag used to set the code128 max length property.	<META HTTP-Equiv="i2of5-maxlength" content="[parameter]">	nn 0 to 55	<META HTTP-Equiv="i2of5-maxlength" Content="30">
i2of5-converttoean13	The i2of5-converttoean13 META Tag is an action tag used to set the i2of5 converttoean13 property	<META HTTP-Equiv="i2of5-converttoean13" content="[parameter]">	True - enables the property False - disables the property	<META HTTP-Equiv="i2of5" Content="enabled"> <META HTTP-Equiv="ean13" Content="enabled"> <META HTTP-Equiv="i2of5-converttoean13" Content="true">
Discrete 2 of 5				
D2of5	The D2of5 META Tag is an action tag used to enable or disable the D2of 5 decoder.	<META HTTP-Equiv="D2of5" content="[parameter]">	Enabled - enables the decoder Disabled - disables the decoder	<META HTTP-Equiv="D2of5" Content="enabled">
D2of5 -Redundancy	The D2of5 -Redundancy META Tag is an action tag used to set the D2of5 redundancy property	<META HTTP-Equiv="D2of5-redundancy" content="[parameter]">	True - enables the property False - disables the property	<META HTTP-Equiv="D2of5-redundancy" Content="true">
D2of5 -MinLength	Sets the minimum number of characters to nn	<META HTTP-Equiv="D2of5-minlength" content="[parameter]">	nn 0 to 55	<META HTTP-Equiv="D2of5-minlength" Content="30">
D2of5 -MaxLength	Sets the maximum number of characters to nn	<META HTTP-Equiv="D2of5-maxlength" content="[parameter]">	nn 0 to 55	<META HTTP-Equiv="D2of5-maxlength" Content="30">
Codabar				
Codabar	The Codabar META Tag is an action tag used to enable or disable the Codabar decoder By default enabled	<META HTTP-Equiv="Codabar" content="[parameter]">	Enabled - enables the decoder Disabled - disables the decoder	<META HTTP-Equiv="Codabar" Content="enabled">

Table C-1 *Custom Meta Tags (Continued)*

Symbology Property	Description	Custom Tags	Parameters	Example:
Codabar-Redundancy	The Codabar-Redundancy META Tag is an action tag used to set the Codabar redundancy property	<META HTTP-Equiv="Codabar-redundancy" content="[parameter]">	True - enables the property False - disables the property	<META HTTP-Equiv="Codabar-redundancy" Content="true">
Codabar-MinLength	The Codabar-MinLength META Tag is an action tag used to set the Codabar min length property.	<META HTTP-Equiv="Codabar-minlength" content="[parameter]">	nn 0 to 55	<META HTTP-Equiv="Codabar-minlength" Content="30">0">
Codabar-MaxLength	The Codabar-MaxLength META Tag is an action tag used to set the Codabar max length property.	<META HTTP-Equiv="Codabar-maxlength" content="[parameter]">	nn 0 to 55	<META HTTP-Equiv="Codabar-maxlength" Content="30">
Codabar-NotisEditing	The Codabar-NotisEditing META Tag is an action tag used to set the codabar not is editing property.	<META HTTP-Equiv="codabar-notiseditiing" content="[parameter]">	True - enables the property False - disables the property	<META HTTP-Equiv="codabar-notiseditiing" Content="true">
Codabar-ClsiEditing	The Codabar-ClsiEditing META Tag is an action tag used to set the codabar clsiediting property.	<META HTTP-Equiv="codabar-clsiediting" content="[parameter]">	True - enables the property False - disables the property	<META HTTP-Equiv="codabar-clsiediting" Content="true">
UPC-E0				
UPC E0	The upce0 META Tag is an action tag used to enable or disable the upce0 decoder. By default enabled.	<META HTTP-Equiv="upce0" content="[parameter]">	Enabled - enables the decoder Disabled - disables the decoder	<META HTTP-Equiv="upce0" Content="enabled">
upce0-ReportCheckDigit	The upce0-ReportCheckDigit META Tag is an action tag used to set the upce0 reportcheckdigit property.	<META HTTP-Equiv="upce0-reportcheckdigit" content="[parameter]">	True - enables reporting of the check digit False - disables reporting of the check digit	<META HTTP-Equiv="upce0-reportcheckdigit" Content="true">
upce0-preamble	The upce0-preamble META Tag is an action tag used to set the upce0-preamble property	<META HTTP-Equiv="upce0-preamble" content="[parameter]">	PREAMBLE_NONE no preamble. PREAMBLE_SYSTEM_CHAR system character preamble PREAMBLE_COUNTRY_AND_SYSTEM_CHARS both system and country code preamble	<META HTTP-Equiv="upce0-preamble" Content="PREAMBLE_COUNTRY_AND_SYSTEM_CHARS">

Table C-1 *Custom Meta Tags (Continued)*

Symbology Property	Description	Custom Tags	Parameters	Example:
upce0-ConvertToupca	The upce0-ConvertToupca META Tag is an action tag used to set the upce0-converttoupca property.	<META HTTP-Equiv="upce0-convertt oupca" content="[parameter]">	True - enables conversion of UPC-E0 bar codes to UPC-A False - disables conversion of UPC-E0 bar codes to upca	<META HTTP-Equiv="upce0-convertt oupca" Content="true">
UPC-E1				
UPC e1	The upce1 META Tag is an action tag used to enable or disable the upcE1 decoder. By default disabled.	<META HTTP-Equiv="upce1" content="[parameter]">	Enabled - enables the decoder Disabled - disables the decoder	<META HTTP-Equiv="upce1" Content="enabled">
upce1-ReportCheckDigit	The upce1-ReportCheckDigit META Tag is an action tag used to set the upce1 reportcheckdigit property.	<META HTTP-Equiv="upce1-reportch eckdigit" content="[parameter]">	True - enables reporting of the check digit. False - disables reporting of the check digit.	<META HTTP-Equiv="upce1-reportch eckdigit" Content="true">
upce1-preamble	The upce1-preamble META Tag is an action tag used to set the upce1-preamble property	<META HTTP-Equiv="upce1-preambl e" content="[parameter]">	PREAMBLE_NON E no preamble. PREAMBLE_SYST EM_CHAR system character preamble. PREAMBLE_COU NTRY_AND_SYST EM_CHARS both system and country code preamble.	<META HTTP-Equiv="upce1-preambl e" Content="PREAMBLE_COU NTRY_AND_SYSTEM_CHA RS">
upce1-ConvertToupca	The upce1-ConvertToupca META Tag is an action tag used to set the upce1-converttoupca property.	<META HTTP-Equiv="upce1-convertt oupca" content="[parameter]">	True - enables conversion of UPC-E1 bar codes to UPC-A False - disables conversion of UPC-E1 bar codes to UPC-A	<META HTTP-Equiv="upce1-convertt oupca" Content="true">
UPC-A				
UPC A	The upca META Tag is an action tag used to enable or disable the upcA decoder. By default enabled.	<META HTTP-Equiv="upca" content="[parameter]">	Enabled - enables the decoder Disabled - disables the decoder	<META HTTP-Equiv="upca" Content="enabled">

Table C-1 *Custom Meta Tags (Continued)*

Symbology Property	Description	Custom Tags	Parameters	Example:
upca-preamble	The upca-preamble META Tag is an action tag used to set the upca-preamble property	<META HTTP-Equiv="upca-preamble" content="[parameter]">	PREAMBLE_NONE no preamble. PREAMBLE_SYSTEM_CHAR system character preamble PREAMBLE_COUNTRY_AND_SYSTEM_CHARS both system and country code preamble	<META HTTP-Equiv="upca-preamble" Content="PREAMBLE_COUNTRY_AND_SYSTEM_CHARS">
EAN-8				
EAN8	The ean8 META Tag is an action tag used to enable or disable the ean8 decoder. By default enabled	<META HTTP-Equiv="ean8" content="[parameter]">	Enabled - enables the decoder Disabled - disables the decoder	<META HTTP-Equiv="ean8" Content="enabled">
ean8-converttoean13	The ean8-converttoean13 META Tag is an action tag used to set the ean8 converttoean13 property.	<META HTTP-Equiv="ean8-converttoean13" content="[parameter]">	True - Enables the property False - disables the property	<META HTTP-Equiv="ean8-converttoean13" Content="true">
EAN-13				
EAN13	The ean13 META Tag is an action tag used to enable or disable the ean13 decoder. By default enabled	<META HTTP-Equiv="ean13" content="[parameter]">	Enabled - enables the decoder Disabled - disables the decoder	<META HTTP-Equiv="ean13" Content="enabled">
Code 11				
Code11	The Code11 META Tag is an action tag used to enable or disable the code11 decoder. By default disabled.	<META HTTP-Equiv="code11" content="[parameter]">	Enabled - enables the decoder Disabled - disables the decoder	<META HTTP-Equiv="code11" Content="enabled">
Code11-Redundancy	The Code11-Redundancy META Tag is an action tag used to set the code11 redundancy property	<META HTTP-Equiv="code11-redundancy" content="[parameter]">	True - Enables the property False - disables the property	<META HTTP-Equiv="code11-redundancy" Content="true">
Code11-ReportCheckDigit	The Code11-ReportCheckDigit META Tag is an action tag used to set the code11 reportcheckdigit property.	<META HTTP-Equiv="code11-reportcheckdigit" content="[parameter]">	True - Enables reporting of the check digit False - disables reporting of the check digit	<META HTTP-Equiv="code11-reportcheckdigit" Content="true">

Table C-1 *Custom Meta Tags (Continued)*

Symbology Property	Description	Custom Tags	Parameters	Example:
Code11-CheckDigitCount	The Code11-CheckDigitCount META Tag is an action tag used to set the code11-checkdigitcount property.	<META HTTP-Equiv="code11-checkdigitcount" content="[parameter]">	**code11_no_check_digit** - verify no check digits **code11_one_check_digit -** verify one check digit **code11_two_check_digit** - verify two check digits	<META HTTP-Equiv="code11-checkdigitcount" Content="code11_one_check_digit">
Trioptic Code 39				
Trioptic39	The Trioptic39 META Tag is an action tag used to enable or disable the Trioptic39 decoder. By default disabled.	<META HTTP-Equiv="trioptic39" content="[parameter]">	Enabled - enables the decoder Disabled - disables the decoder	<META HTTP-Equiv="trioptic39" Content="enabled">
Trioptic39-Redundancy	The Trioptic39-Redundancy META Tag is an action tag used to set the Trioptic39 redundancy property	<META HTTP-Equiv="trioptic39-redundancy" content="[parameter]">	True - Enables the property False - disables the property	<META HTTP-Equiv="trioptic39-redundancy" Content="true">
GS1 DataBar				
GS1 DataBar-14	The Rss14 META Tag is an action tag used to enable or disable the Rss14 decoder. By default disabled.	<META HTTP-Equiv="rss14" content="[parameter]">	Enabled - enables the decoder Disabled - disables the decoder	<META HTTP-Equiv="rss14" Content="enabled">
GS1 DataBar Limited	The RssLim META Tag is an action tag used to enable or disable the RssLim decoder. By default disabled.	<META HTTP-Equiv="rsslim" content="[parameter]">	Enabled - enables the decoder Disabled - disables the decoder	<META HTTP-Equiv="rsslim" Content="enabled">
GS1 DataBar Expanded	The Rssexp META Tag is an action tag used to enable or disable the Rssexp decoder. By default disabled.	<META HTTP-Equiv="rssexp" content="[parameter]">	Enabled - enables the decoder Disabled - disables the decoder	<META HTTP-Equiv="rssexp" Content="enabled">
Webcode				
Webcode	The Webcode META Tag is an action tag used to enable or disable the Webcode decoder. By default disabled.	<META HTTP-Equiv="webcode" content="[parameter]">	Enabled - enables the decoder Disabled - disables the decoder	<META HTTP-Equiv="webcode" Content="enabled">

Configuring Symbologies During the Login Process

If the default login process of the UCA server is used, symbologies can be configured by embedding the appropriate META tags in the file scanner_symbologies.config located in C:\Inetpub\wwwroot\Motorola\UCA\Client\.

The configuration in these META tags are applied to the device when the user logs in. The administrator should ensure that the meta tags included in this file are valid and formatted correctly. A user login error on the device results if the format is not correct.

Configuring Symbologies in a Business Application

Every business application (Web page) can include META tags to configure the symbologies.

NOTE When the symbology is configured, it remains until the user warm boots or cold boots the device.

Example:

To change the minimum length of symbology *I 2 of 5* to 10 use the following meta tag in the html file in the header section:

<META HTTP-Equiv="i2of5-minlength" Content="10">

NOTE For sample business applications see *Appendix I, Business Applications Supplement*.

Appendix D Deployment Example - UCA Server Install & Basic User Setup

Overview

This deployment example is a step-by-step exercise that takes a developer through the installation of a UCA server, its configuration and the deployment of CA50s, in addition to the topics listed below:

- Installing and configuring UCA server production version software
- Setting up the CA50 device
  - Generating Rapid Deployment bar codes using MSP software
  - Creating an XML profile from UCA Profile Builder (which configures the device when it is loaded from the UCA server to the CA50 device)
- Populating UCA server files and databases with site/user specific information
- Generating an employee login bar code
- Deploying a CA50
- Logging an employee into the UCA server.

Upon completing this exercise, the user can use the CA50 to communicate via voice calls and walkie-talkie communication.

This exercise does not enable price lookup. For a self contained pre-built demo that enables price lookup, voice calls and walkie-talkie communication, see *Appendix E, Self Contained Pre-Built Demo*.

NOTE 1. For a complete checklist outlining all the development and deployment activities required to implement a CA50 solution, see the *Development and Deployment Checklist on page 2-1*.

2. For a table listing the most commonly requested topics within this manual, see the Quick Startup Guide on the last page of the manual.

Outline of Steps to Perform

To get the CA50 Demo Solution running, perform the following steps:

1. Review the required settings for this exercise (*page D-3*).
2. Set up the static IP on the server hardware (*page D-4*).
3. Install the UCA server software (*page D-4*).
4. Validate the successful server installation on the laptop (*page D-5*).
5. Create an XML profile (*page D-6*).
6. Populate the UCA server database (*page D-11*).
7. Create a login bar code (*page D-15*).
8. Set up an RF network (*page D-16*).
9. Use MSP to create CA50 Rapid Deployment bar codes (*page D-16*).
10. Deploy CA50s (*page D-18*).

Step 1. Required Site Settings for this Exercise

Wireless Network Site Settings

- Wireless ESSID: UCA
- Wireless Encryption: None (open)

CA50 and UCA Server Settings

- UCA Server Static IP Address: 192.168.0.253
- Store #ID: 0002
- CA50 User Names: User1 and User2
- External UserID Bar Code: 1001 for User1 and 1002 User2
- CA50 User Role: Shoe Sales "They sell shoes on the showroom floor"
- CA50 User Group: Footwear
- Phone Extensions for Group 10 to 13
- Walkie-Talkie Channel: Channel 3 Only
- User Backlight Setting: Level 3
- Installshield SA Password: SQLisGreat01
- Installshield SQL Database Login ID: Motorola02
- Installshield SQL Database Password: UCAisGreat02
- Base XML Profile to Create Footwear.xml
- PBX Not required. PBX is not used in this exercise.

Business Application Settings

- Business Web Application URL: http://192.168.0.253/Footwear/Footwear.aspx.

NOTE This application must be created by a Web developer and is not part of the UCA installation. This is used for example purposes only.

NOTE For sample business applications see *Appendix I, Business Applications Supplement*.

Step 2. Setup Static IP on Server Hardware

UCA Server Static IP Address: 192.168.0.253

Subnet Mask: 255.255.255.0

To do this in Windows 2003:

1. Go to **Start -> Control Panel -> Network Connections**, and click on **Local Area Connection**.
2. In the *Local Area Connection Status* window, select **Properties**.
3. In the *Local Area Connection Properties* window, highlight the **Internet Protocol (TCP/IP)** option and click **Properties**.
4. Select the **Use the following IP address** button and enter the values above.

Important Notes:

- The CA50 device must know where the UCA server IP address is for configuration via Rapid Deployment bar codes. For this to occur, set the UCA server hardware to the UCA server static IP address above (192.168.0.253).
- Ensure this IP address can be pinged through the wireless network using the CA50 for the ping testing (see *UCAPing on page 4-27*). Ensure firewalls are disabled, or allow access to this IP address from the wireless network.

Step 3. Install UCA Server Software

- Confirm that the PC meets the hardware requirements (see *Minimum Server Hardware Requirements on page 6-7*).
- Confirm that the PC meets the OS and software requirements (see *Server Software Requirements. on page 6-7*).
- Load UCA server software on the PC (see *Installing UCA Server Software on page 6-10*).
  - During the UCA server installation process, use the following login information:

    SA password = SQLisGreat01$ (for additional information, see *page 6-13*, Step 10)

    UCA database login ID = Motorola02 (for additional information, see *page 6-13*, Step 15)

    UCA database password = UCAisGreat02 (for additional information, see *page 6-15*, Step 15).

IMPORTANT The installation takes approximately 15 to 20 minutes, depending on whether or not .NET Framework is already loaded, and the computer processor speed. During the installation process, the Installshield progress bar may not always appear to be moving. This is normal behavior and does not mean the installation process stopped.

Step 4. Validate Successful Server Installation on Laptop

To verify the installation and configuration of the UCA server software follow the process documented in *Validate Successful Server Installation and Proper Performance on page 6-45*. When performing the validation process, use the default Web Console user login information:

Launch the UCA server web console from the desktop shortcut.

- Default User ID: *admin*
- Password: *admin*

IMPORTANT Ensure to enter all settings called out on *page 6-45* exactly as shown.

IMPORTANT At the initial login to the Web Console (for details see *page 9-3*), it is required to use the default Web Console administrator *Username* and *Password*:

- *Factory Default Web Console Username*: *admin*
- *Factory Default Web Console Password*: *admin*

It is strongly suggested that the default administrator *Username* and *Password* be changed after the initial login.

Note that the *Username* and *Password* entered during the UCA server software installation process, (*step on page 6-13*) are for SQL database access and should not be mistaken with the default Web Console administrator Username and Password shown above.

Step 5. Create an XML Profile Called "Footwear.xml"

A *base XML profile* is needed to build the baseline settings for CA50 users. When a CA50 user logs in to the UCA server, the UCA server searches the database for the user *Role* and/or *Group* information and selects which base XML profile to use. The UCA server then fills in the remainder of the base XML profile settings from internal databases and files and sends back to the CA50 the completed base XML profile. This information is processed by the CA50 client to setup the CA50.

For this exercise, the base XML profile is Footwear.xml and the two users, User1 and User2, use this XML profile when they log in to the UCA server.

To start:

1. Use the Profile Builder and open c:\Program Files\Motorola\UCA Server\Profiles\bpaSales.xml

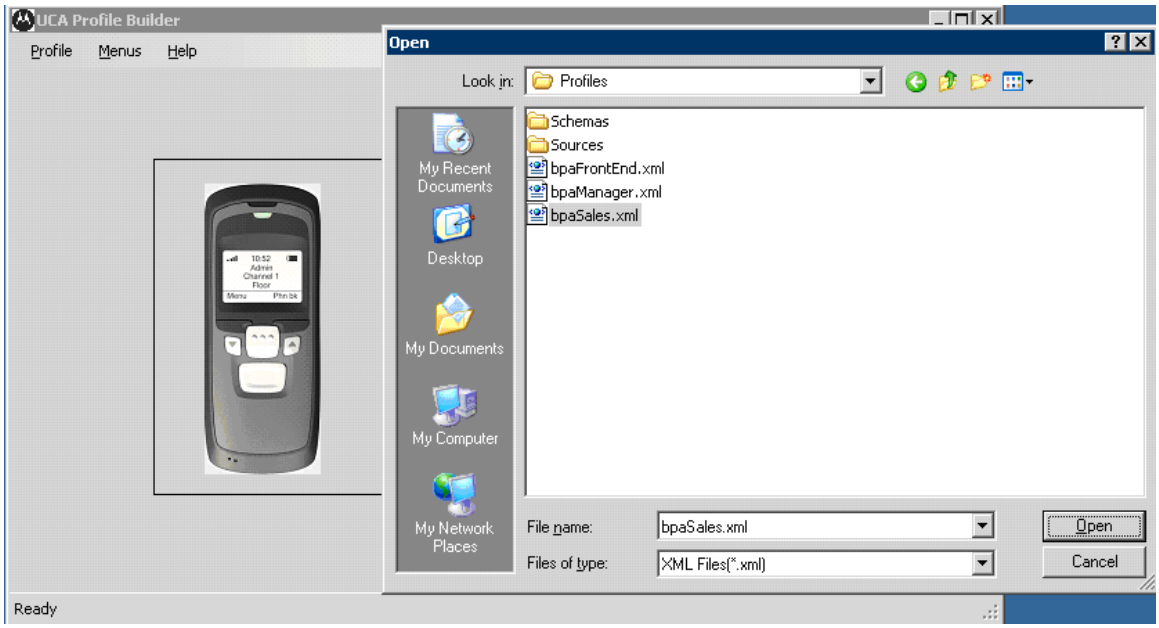


Figure D-1 *UCA Server Log-in*

2. Save bpaSales.xml as Footwear.xml. Keep Footwear.xml in the same directory as bpaSales.xml. Do not delete bpaSales.xml. It is used by other applications.l

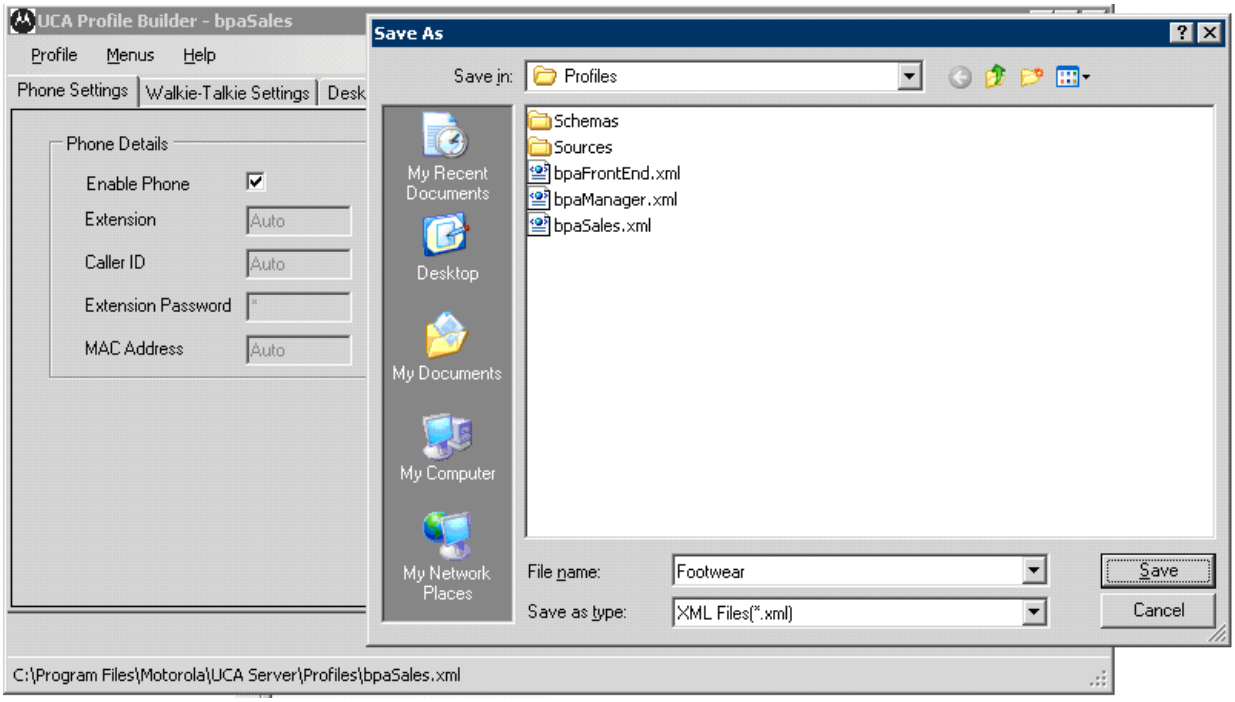


Figure D-2 *UCA Server Profile Save*

3. Go to the *Walkie-Talkie Settings* tab, select the check box for channel number 3 and click **Remove** to delete.l

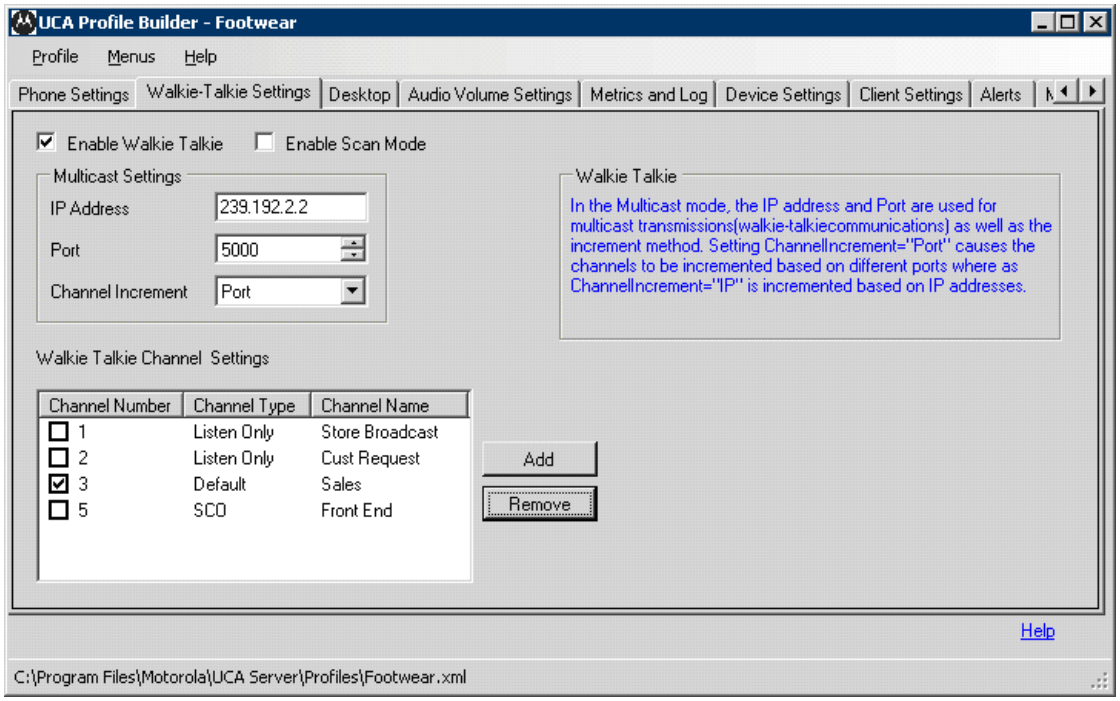


Figure D-3 *Channel Number Check*

4. Click **Add**, to add the channel details. Enter the information.

   Channel Number: 3

   Channel type : Default

   Channel Name : Footwear.l

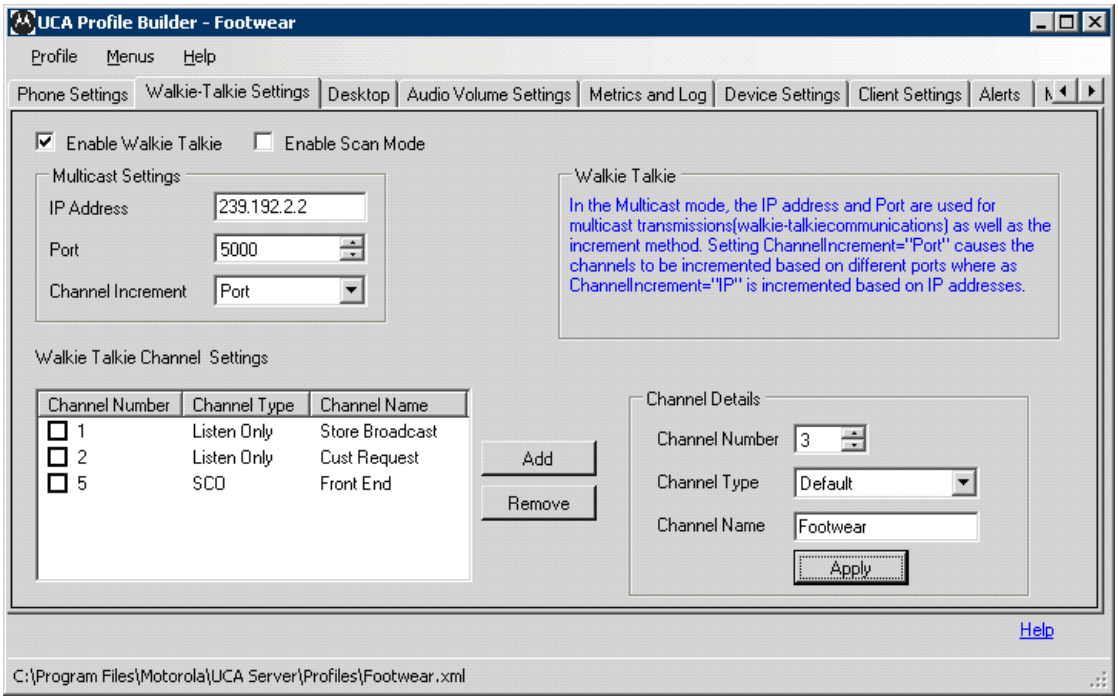


Figure D-4 *Add Channel Details*

5. Click **Apply** to save the channel settings.

6. Go to the *Menus and Sub-menus* tab and expand the menus tree. Right click on *AppIns* (Applications) menu and select *Add Server Action*.l

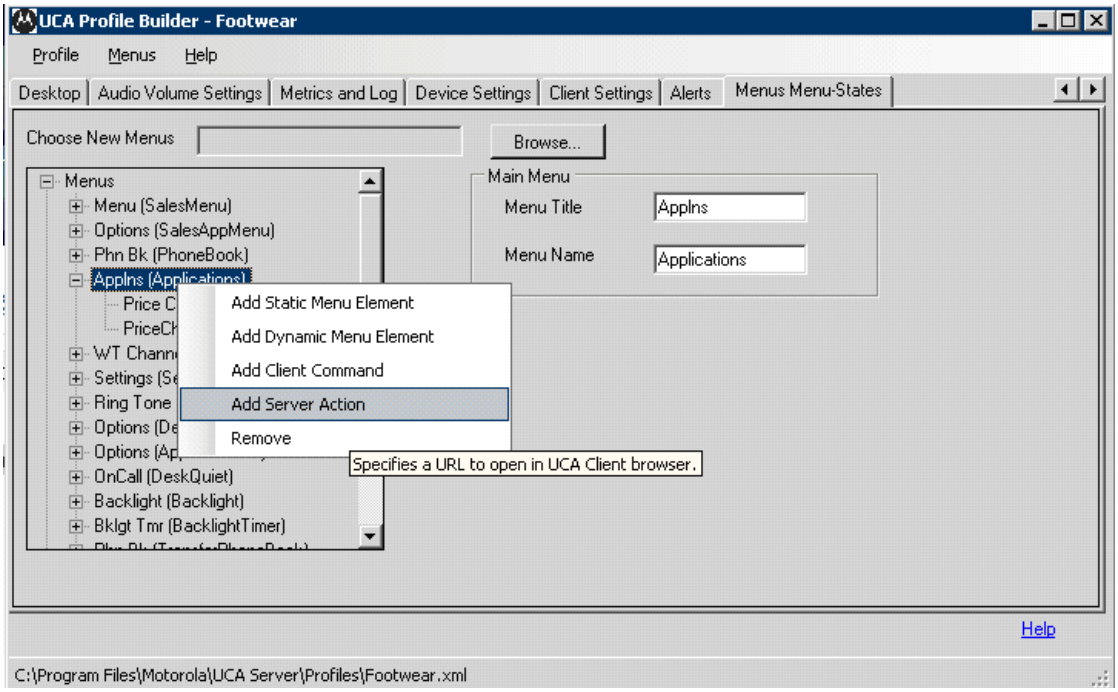


Figure D-5 *Add Server Action*

7. A new sub-menu *ServerAction1* appears under the *AppIns* menu. Select the sub-menu and edit the menu title to *Footwear* and for the URI provide the link as http://192.168.0.253/Footwear/Footwear.aspx. This points to the *Footwear Business Application*.l

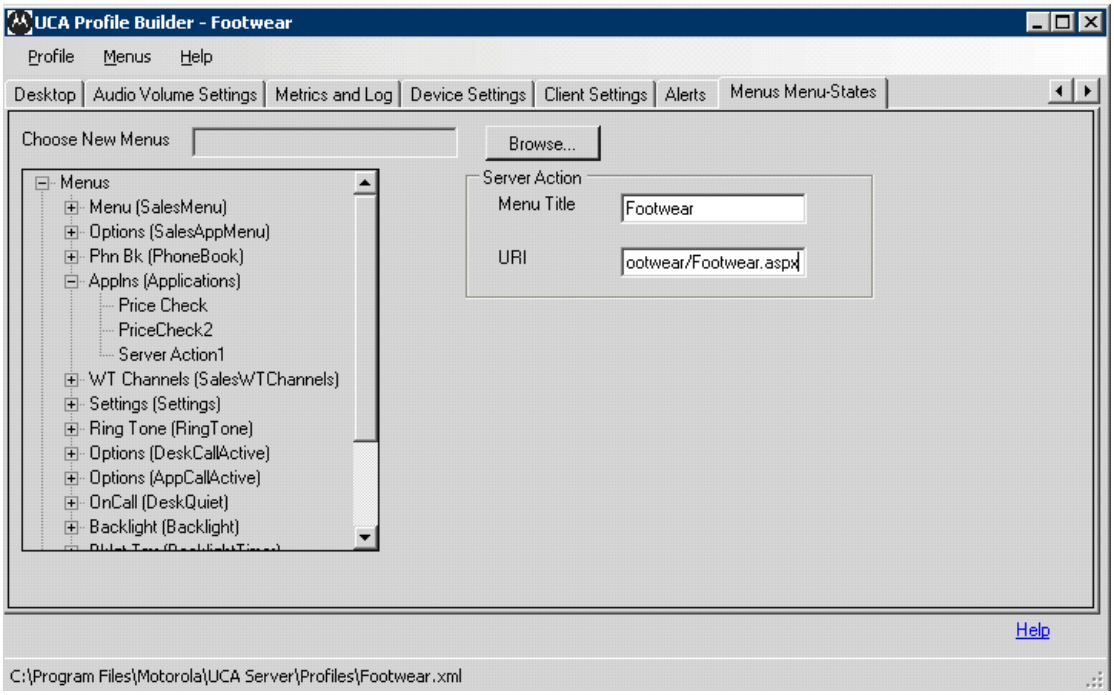


Figure D-6 *ServerAction1*

8. Expand the *WT Channels* (SalesWTChannel) menu. Select *My WT Settings* and change the menu title to *Footwear*.l

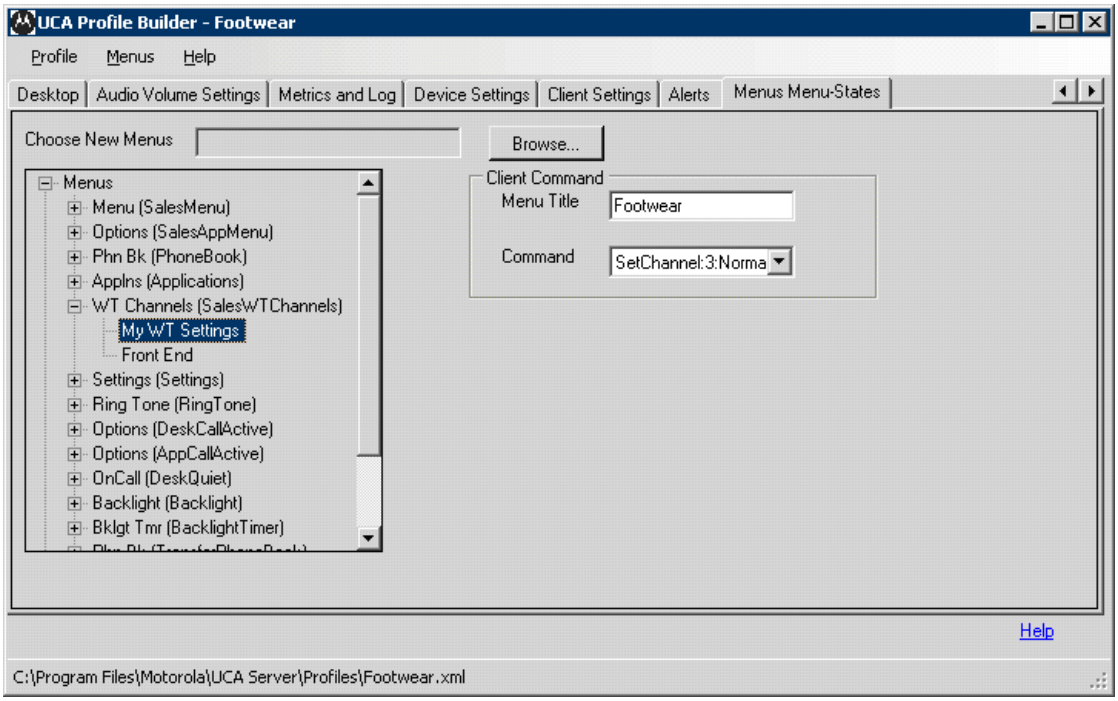


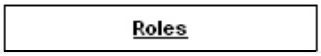
Figure D-7 *WT Settings*

Step 6. Populate the UCA Server Database with 2 Users

User information such as *Roles*, *Groups*, *Group Phone Extensions*, *Base XML Profile Association*, *Name* and *ExternalUser ID Barcode Login* are added to the UCA server databases in the UCA server Web Console GUI.

To add user information:

1. Login to the UCA server Web Console GUI.
   - Go to: http://localhost/Motorola/UCA/UCAConsole/Login.aspx
   - Enter Username: admin
   - Enter Password: admin.
2. Create a “ShoeSales" *Role*.
   - Click

.
   - Click 

.
   - Click

.
   - Enter the field information shown in *Figure D-8* and click **Save**.

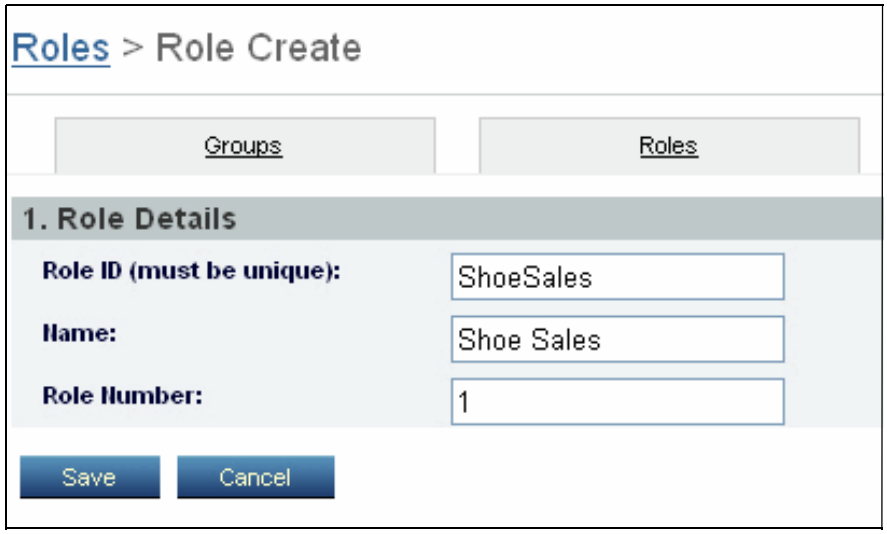
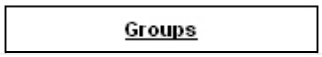


Figure D-8 *Create a Role*

3. Create a “Footwear" *Group*.
   - Click

.
   - Click 

.

- Click

.
- Enter the field information shown in *Figure D-9* and click **Save**.

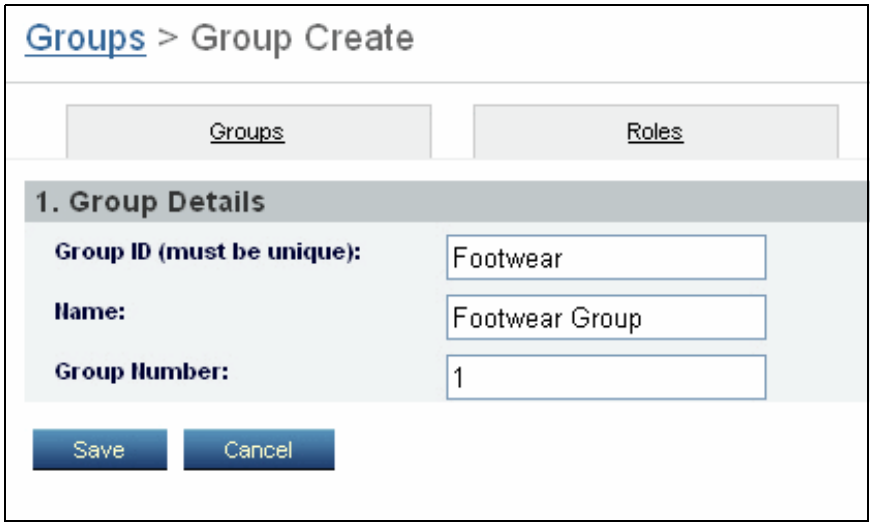
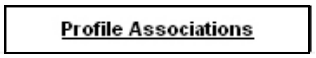


Figure D-9 *Create a Group*

4. Create a Base XML profile association with the *Role* and *Group*.

   When the CA50 user logs into the UCA server, their role and/or group must be associated to a base XML profile. For this exercise, Footwear.xml was created for the association. (See *Step 5. Create an XML Profile Called "Footwear.xml" on page D-6*.)

   - Click

.
   - Click 

.
   - Click

.

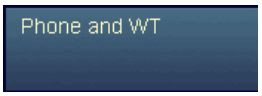
- Enter the field information shown in *Figure D-10* and click **Save**.

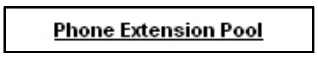
Figure D-10 *Profile Association*

5. Create *Phone Book Extensions* based on *Group.*

   a. Create four phone extensions (extensions 10,11,12 and 13) for the *Footwear Group*. When the users in the *Footwear Group* login, they get one of these extensions.

   To create phone book extensions:

   - Click 

.
   - Click 

.
   - Click

.
   - Enter the field information for phone extension 10 shown in *Figure D-11* and click **Save**.

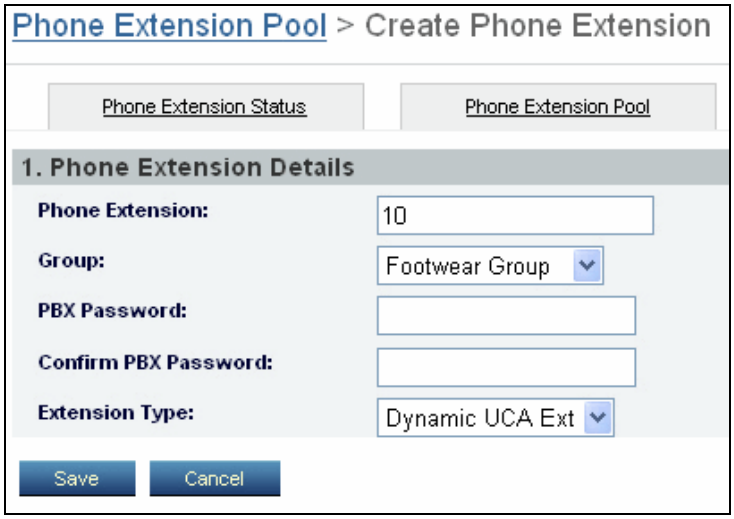
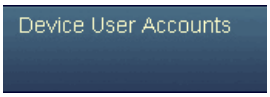


Figure D-11 *Create Phone Extension*

   b. Repeat this process and create phone extensions for extensions 11,12, and 13.

6. Create two users, User1 and User2, each with a *Shoe Sales Role* in the *Footwear Group*.
   - User1 is using the *External UserID 1001*.
   - User2 is using the *External UserID 1002*.

   To create User1 and User2:

   - Click 

.
   - Click

.
   - Click

.
   - Enter the field information for User1 shown in *Figure D-12* and click **Save**.
     Remember to assign the users to *Role: Shoe Sales* and *Permanent Group: Footwear Group*

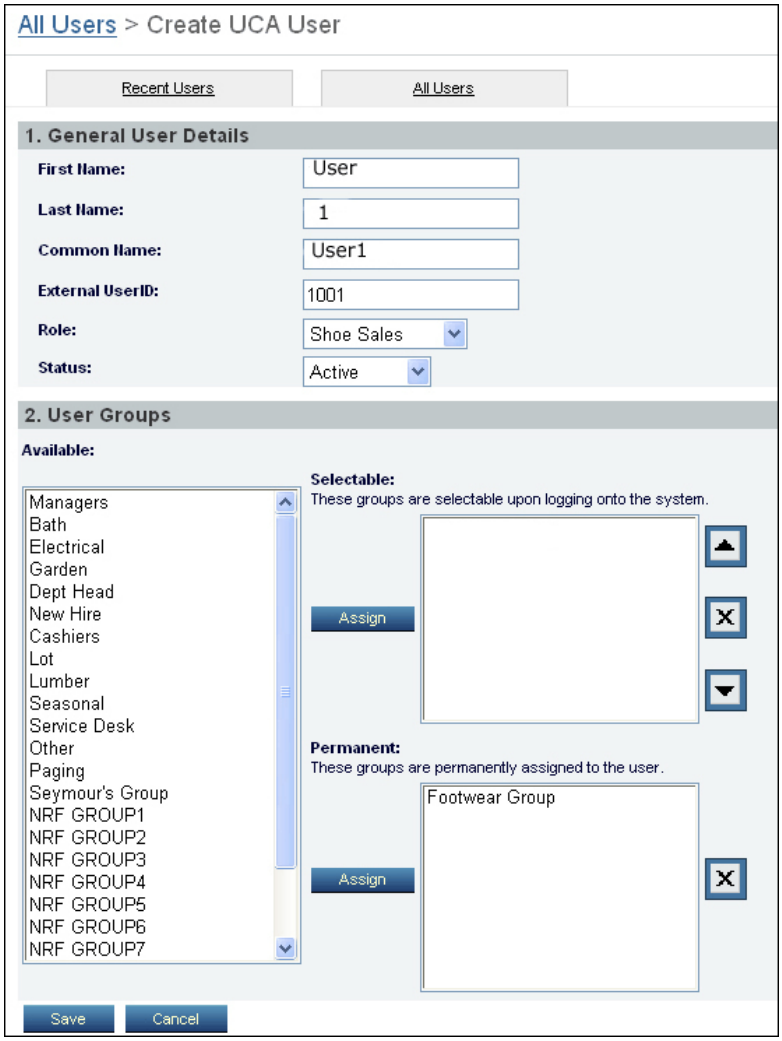


Figure D-12 *Create Users*

   - Repeat the process to create User2 (using *External UserID 1002*).

Step 7. Generate Employee Login Bar Codes

Generate a login bar code(s) using the newly created external User ID. Bar codes can be generated using off-the-shelf third party bar code generating software. This bar code is scanned in Step 8. For additional details on scanning a login bar code, see *page 4-3*.

Step 8. RF Network

Two options exist for using an RF network.

- Option A: Use your existing RF network.
- Option B: Set up a simple RF network using a WS2000. For details, see *Appendix E, Self Contained Pre-Built Demo* (*Step 5. Configure the RF Hardware on page E-5*).

The CA50 scans for access point transmissions on the RF channels specified for the country in which it is used. In the Americas, when the 802.11b/g radio is enabled the CA50 defaults to scanning the three non-overlapping radio channels 1, 6 and 11, which correspond to frequencies 2412 Mhz, 2437 Mhz and 2462 Mhz respectively. When the 802.11a radio is enabled, the CA50 scans for access points on all the country-appropriate channels in the 5.0Ghz range.

To designate specific channels to scan, a channel mask can be applied using the following registry keys where Freq1, Freq2 and Freq3 correspond to standard 802.11 channel frequencies in Mhz.

```
[HKEY_LOCAL_MACHINE\Comm\JEDI10_1\PARMS]
"ChannelMask"="Freq1 Freq2 Freq3 …"
[HKEY_LOCAL_MACHINE\Comm\PW\JEDI10_1\PARMS]
"ChannelMask"=" Freq1 Freq2 Freq3 …"
```

In the following example, the frequencies refer to 802.11a channels 36, 40, 44 and 48. The registry keys limit scanning to these channels.

```
[HKEY_LOCAL_MACHINE\Comm\JEDI10_1\PARMS]
"ChannelMask"="5180 5200 5220 5240"
[HKY_LOCAL_MACHINE\Comm\PW\JEDI10_1\PARMS]
"ChannelMask"="5180 5200 5220 5240"
```

Step 9. Use MSP 3.1 to Prepare CA50 Devices for Use

NOTE Load MSP software using the MSP installation instructions.

IMPORTANT When installing MSP 3.2.1, the UCA settings screen includes a *Language* drop-down list. Select the appropriate language to localize the CA50 device. The language selected must match the language selected when installing UCA server software. If the language selections does not match, the UCA server software language (selected during the UCA server software installation) the localization of device happens after the user login.

Use MSP 3.2.1 to prepare the CA50 devices for use.To generate Rapid Deployment bar codes to connect the CA50 device to the wireless network and establish communication with the UCA server, ensure all settings (RF wireless network settings, IP address of the UCA server, etc.) are at hand:

- ESSID = UCA
- Encryption = None (Open)
- UCA Server IP= 192.168.0.253
- Store #: 0002.

Prior to creating RD bar codes for the CA50, see *Chapter 12, Staging and Deploying CA50s* for detailed information about MSP settings and MSP staging profiles, including:

- *CA50 Parameters Required to Generate RD Bar Codes on page 12-2*
- *UCA.settings.xml File for Configuring the UCA Client Application on page 12-2*

- *Creating the Agent 30 settings definition on page 12-4*
- *Creating the UCA settings definition on page 12-4*
- *Creating an MSP bundle on page 12-5*
- *Creating an MSP staging profile on page 12-5*.

Create RD Bar Codes for the CA50

To create RD bar codes for the CA50 in the *Staging Profiles* window:

1. Click the *Staging* tab.
2. Click *Staging Profiles*.
3. Under *Profile ID,* on the *CA50.Stage* line, click *Create* under *Barcode*.
4. On the Generate RD Barcode Sheets pop up window, select *Linear* for barcode type and click **Generate** to generate the barcodes.

Step 10. Deploy CA50s

Prepare the CA50: Insert a charged battery into the device (see *page 3-4*, *3-8*).

Deploy CA50s at the Site Location

- Warm boot to CA Menu (see *Performing a Warm Boot on page 4-17*). To go to CA Menu, press and hold the left soft key.
- Launch the Rapid Deployment Client on the CA50 from the CA50 Menu (see *CA Menu on page 3-12*).
- Scan the RD bar codes created in step *Step 7. Generate Employee Login Bar Codes on page D-15*.
- The CA50 device is ready to use. Press **OK** to continue.

Validate that the CA50 was Deployed Properly

- From the CA Menu use the **Up** and **Down** keys to scroll and select *Warm Boot* to boot the CA50.

NOTE To automate the warm boot of the device and automatically launch the UCA client application, see *page 12-2*.

- Wireless signal strength bars appear in the upper left corner of the CA50 display. This shows connectivity.
- The UCA client application on the CA50 launches and negotiates/registers with UCA server.
- The CA50 displays *Scan bar code to log in.*

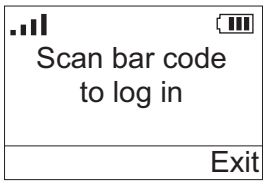


Figure D-13 *Login Prompt*

Employee Logs in to the UCA Server

- Scan external UserID bar code (these bar codes were generated in Step 5 on *page D-15*).
  - 1001 for User1
  - 1002 for User2.
- The UCA server validates the user ID.
- If a valid UserID is found, the UCA server sends the completed base XML profile back to the CA50 user device. See step *Step 6. Populate the UCA Server Database with 2 Users on page D-11* and step *Step 5. Create an XML Profile Called "Footwear.xml" on page D-6*.
- The CA50 user should see their user name and phone book extension on the CA50 display.

NOTE For a fully functional Demo Solution that enables price lookup, voice calls and walkie-talkie communication, see *Appendix E, Self Contained Pre-Built Demo*.

Go to Menu>Applications, to see the application by name *Footwear* was created in using profile builder. Launching it hits the URI (http://192.168.0.253/Footwear/Footwear.aspx) that was assigned when creating the menu.

From desktop, navigate to *Menu>WT Channels*, to see the channel *Footwear* displayed. This was created using UCA Profile builder, set to listen on channel 3.

Appendix E Self Contained Pre-Built Demo

Overview

The CA50 Demo is a self-contained pre-built solution that allows a user to quickly and easily demonstrate all major CA50 and UCA server functionality, such as:

- Walkie-talkie communication
- Voice calls
- Price lookup (business application).

This demo includes **ALL** the documentation and bar codes required to configure and demonstrate walkie-talkie communication, voice calls and price lookup functionality.

Additionally, to enable targeted custom demonstrations, instructions are provided on how to modify the Demo Solution to enable the login of new users.

For developers, the Demo Solution is a bundling of capabilities, such as a price lookup reference implementation (business application); voice call and walkie-talkie functionality provided via the UCA client application; and, sample UCA server implementation including user data. In addition, the demo solution includes a mini-database of SKUs to enable price lookup, which is not part of the UCA server software.

NOTE Server hardware is not available from Motorola and must be purchased separately.

NOTE 1. For a complete checklist outlining all the development and deployment activities required to implement a CA50 solution, see the *Development and Deployment Checklist on page 2-1*.

2. For a table listing the most commonly requested topics within this manual, see the Quick Startup Guide on the last page of the manual.

Outline of Steps to Perform

To get the CA50 Demo Solution running, perform the following steps:

1. Review the list of hardware requirements to run the Demo Solution (*page E-2*).
2. Set up the static IP on the server hardware (*page E-4*).
3. Install the UCA server software (*page E-4*).
4. Validate that the UCA server was loaded successfully (*page E-5*).
5. Configure the RF hardware (wireless switch and Access Point) (*page E-5*).
6. Scan the provided RD bar codes to deploy CA50s (*page E-11*).
7. Scan employee login bar codes (*page E-14*).
8. Ready to demonstrate the CA50 voice and scanning functionality (*page E-17*).
9. Optional - Modify the Demo Solution by adding a user (*page E-18*).

Step 1. Hardware Required to Run the Demo Solution

The following hardware is recommended for implementing the CA50 Demo Solution setup:

- Computer to load UCA server software (not available from Motorola)
  - Hardware requirements - for details see *UCA Server Requirements and Setup on page 6-7*
  - OS requirements - for details see *UCA Server Requirements and Setup on page 6-7*
- UCA server software CD
- Wireless infrastructure
  - WS-2000-SME-WWR kit
    - WS-2000-SME-WWR - wireless switch with built-in DHCP server
    - WS2000 firmware version 2.2.3 or higher is recommended
    - WSAP-5110-100-WW - Access Point
    - 50-24000-060 - WS power supply
  - Ethernet cables (quantity = 2)
- CA50 mobile devices (CA5090-0U0LF5KV11R; quantity = 2)
- 8-slot CA50 charging cradle (CHS5000-8000CR)
- CA50/UCA Client & Server Developer Guide (p/n 73E-93914-xx).

Demo Solution System Diagram

IMPORTANT For this setup, the static IP address of the laptop should be set to 192.168.0.253 before installing the UCA server software.

The Demo Solution includes all the bar codes to set up and demonstrate three CA50s working with the hardware called out in *Figure E-1* below.

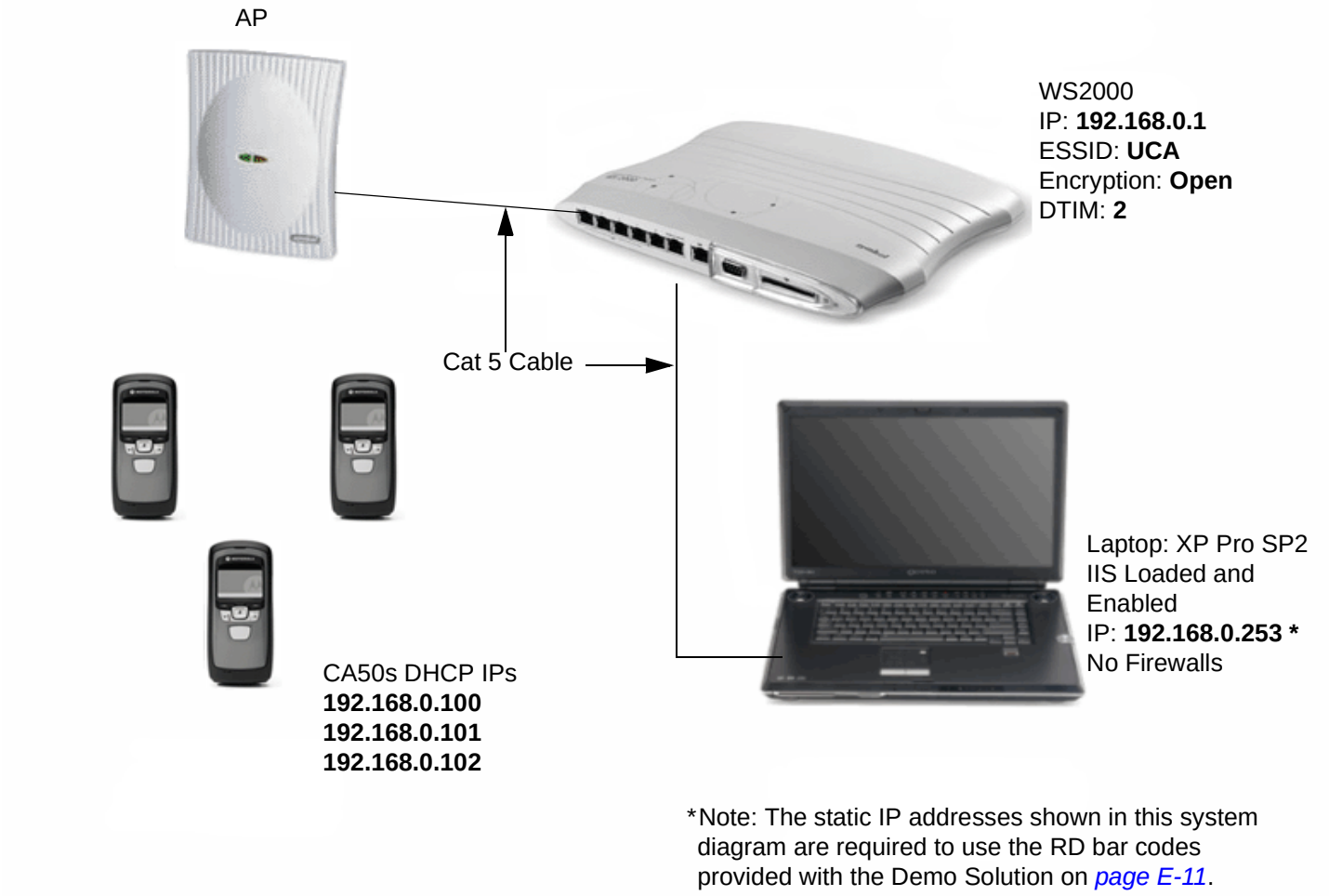


Figure E-1 *Infrastructure*

Step 2. Setup Static IP on Server Hardware

UCA Server Static IP Address: 192.168.0.253

Subnet Mask: 255.255.255.0

To do this in Windows 2003:

1. Go to **Start -> Control Panel -> Network Connections**, and click on **Local Area Connection**.
2. In the *Local Area Connection Status* window, select **Properties**.
3. In the *Local Area Connection Properties* window, highlight the **Internet Protocol (TCP/IP)** option and click **Properties**.
4. Select the **Use the following IP address** button and enter the values above.

Important Notes:

- The CA50 device must know where the UCA server IP address is for configuration via Rapid Deployment bar codes. For this to occur, set the UCA server hardware to the UCA server static IP address above (192.168.0.253).
- Ensure you can ping this IP address through your wireless network using another mobile device or wireless laptop. The CA50 cannot be used for ping testing. Ensure firewalls are disabled, or allow access to this IP address from the wireless network.

Step 3. Install UCA ServerDemo Software

- Confirm that the PC meets the hardware requirements (see *Minimum Server Hardware Requirements on page 6-7*).
- Confirm that the PC meets the OS and software requirements (see *Server Software Requirements. on page 6-7*).
- Load UCA server software on the PC (see *Installing UCA Server Software on page 6-10*).
  - During the UCA server installation process, use the following login information:

    SA password = SQLisGreat01$ (for additional information, see *page 6-13*, Step 10)

    UCA database login ID = Motorola02 (for additional information, see *page 6-15*, Step 15)

    UCA database password = UCAisGreat02 (for additional information, see *page 6-15*, Step 15).

IMPORTANT The installation takes approximately 15 to 20 minutes, depending on whether or not .NET Compact Framework is already loaded, and the computer processor speed. During the installation process, the Installshield progress bar may not always appear to be moving. This is normal behavior and does not mean the installation process stopped.

Step 4. Validate Successful Server Installation on Laptop

To verify the installation and configuration of the UCA server software follow the process documented in *Validate Successful Server Installation and Proper Performance on page 6-45*. When performing the validation process, use the default Web Console user login information:

- Default User ID: *admin*
- Password: *admin*

IMPORTANT Ensure to enter all settings called out on *page 6-45* exactly as shown.

IMPORTANT At the initial login to the Web Console (for details see *page 9-3*), it is required to use the default Web Console administrator *Username* and *Password*:

- *Factory Default Web Console Username*: *admin*
- *Factory Default Web Console Password*: *admin*

It is strongly suggested that the default administrator *Username* and *Password* be changed after the initial login.

Note that the *Username* and *Password* entered during the UCA server software installation process, (*step on page 6-13*) are for SQL database access and should not be mistaken with the default Web Console administrator Username and Password shown above.

Step 5. Configure the RF Hardware

Configure the WS2000 and its AP using the settings listed in *Table E-1*

Table E-1 *WS2000 and AP Configuration*

Field	Setting
LAN Subnet 1	192.168.0.1 <Default>
LAN Subnet 1 Assigned Ports	1 – 6 and WLAN1
Subnet 1 DHCP Server	On
DHCP Range	192.168.0.100 to 192.168.0.150
WLAN 1 ESSID	UCA
WLAN 1 Security Authentication and Encryption	Both Open
Access Ports 802.11 a/b/g Defaults Beacon Settings	DTIM per BSSID to 2
Wireless QoS	Use Voice Prioritization for WLAN1 “Checked”

IMPORTANT WS2000 firmware version 2.2.3 or higher is recommended.

For step-by-step instructions on how to configure the WS2000 and its AP follow the steps below.

1. Connect the CAT 5 cable from the network port in the laptop to the WS2000 port 5 or 6, and provide power to the WS2000.
2. Connect the AP with a Cat 5 cable into WS2000 port 1.
3. Open the web browser to http://192.168.0.1 to view the WS2000 Web Console.

NOTE 1. To access the WS2000 Web Console, your PC must have Java for Windows - Internet Explorer ver. 1.4 or later. This is available at Java.com.

2. It is assumed you are configuring the WS2000 using the PC that hosts the UCA server. If not, make sure the PC is set to DHCP by default.

IMPORTANT If, after loading the Java plug-in, your IE browser is unable to display *Figure E-2*, make sure the ActiveX control is enabled. Go to **Tools > Internet Options > Security** tab and select the **Custom Level** button. Scroll down to **ActiveX controls and plug-ins** and enable the item **Run ActiveX Controls & Plug-ins**.

- Login in to the WS2000 using the factory default username and password:

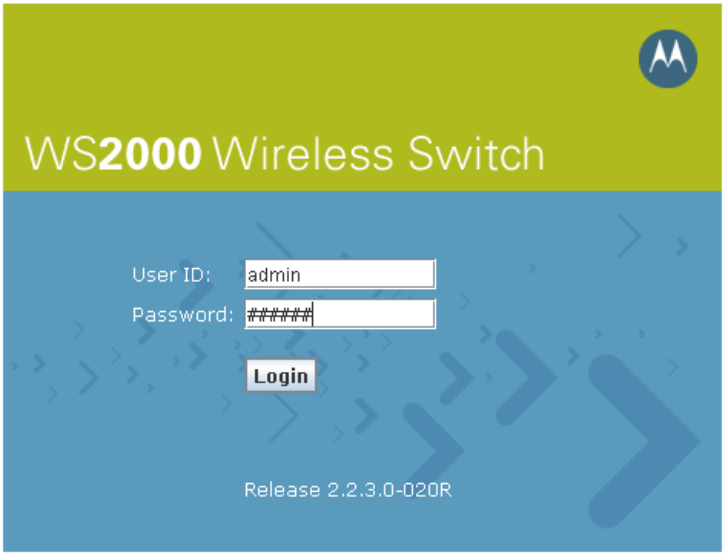


Figure E-2 *WS2000 Wireless Switch Login*

  - Enter the default User ID: *admin*
  - Enter the default Password: *symbol*
  - Click **Login**.
- When prompted to change the WS2000 password, change *symbol* to *Symbol* (change the ‘s’ to upper case ‘S’).
  - New Password: *Symbol*.

4. On the left side of the window under [Network Configuration], click *LAN* > *Subnet1* to configure the WS2000 LAN.

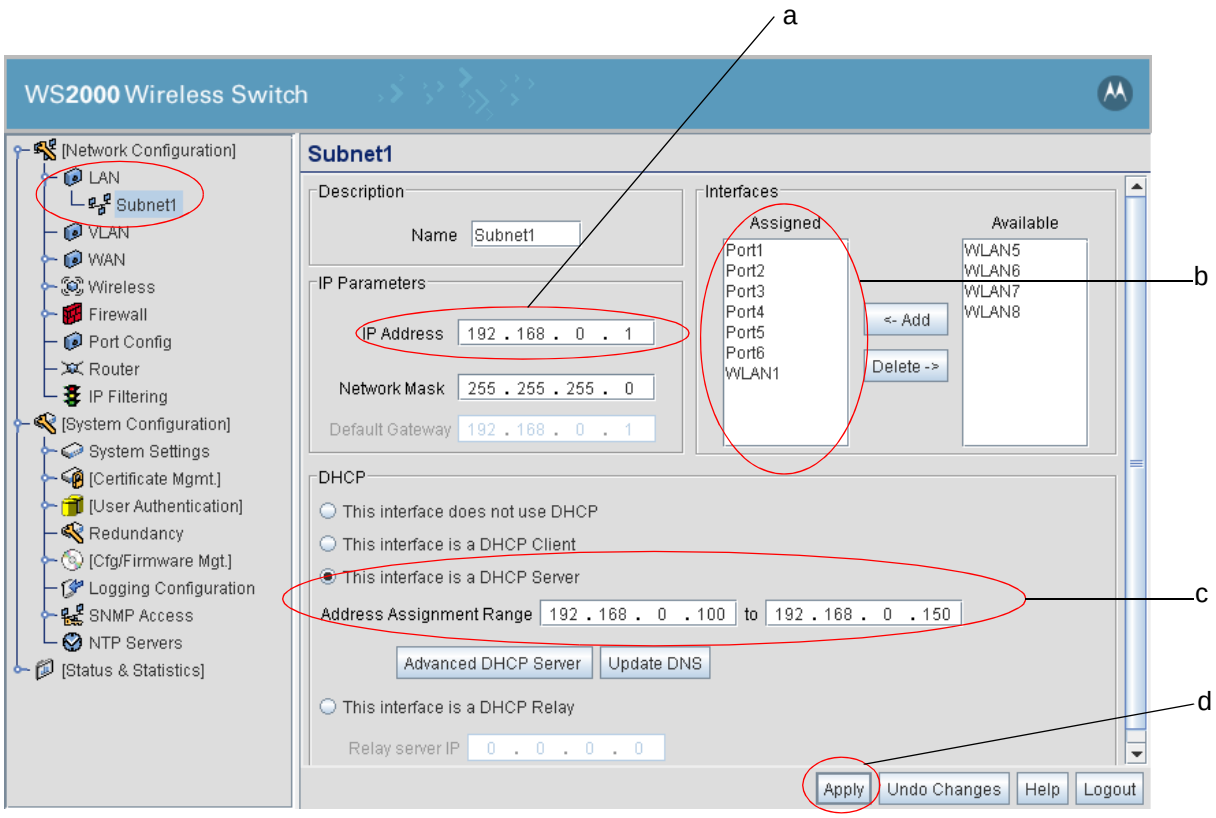


Figure E-3 *Subnet1 Settings*

a. Ensure the static IP address is set to 192.168.0.1 (factory default WS2000 setting).

b. Ensure Ports 1 through 6 and WLAN1 are assigned to Subnet 1. The screen should appear as shown in *Figure E-3*, letter **b**.

c. Enable the DCHP server by clicking the radio button *This Interface is a DHCP Server*. Set the following *Address Assignment Range*: 192.168.0.100 to 192.168.0.150.

d. Click **Apply**.

5. On the left side of the window under [Network Configuration], select *Wireless* > *WLAN1* to configure ESSID.

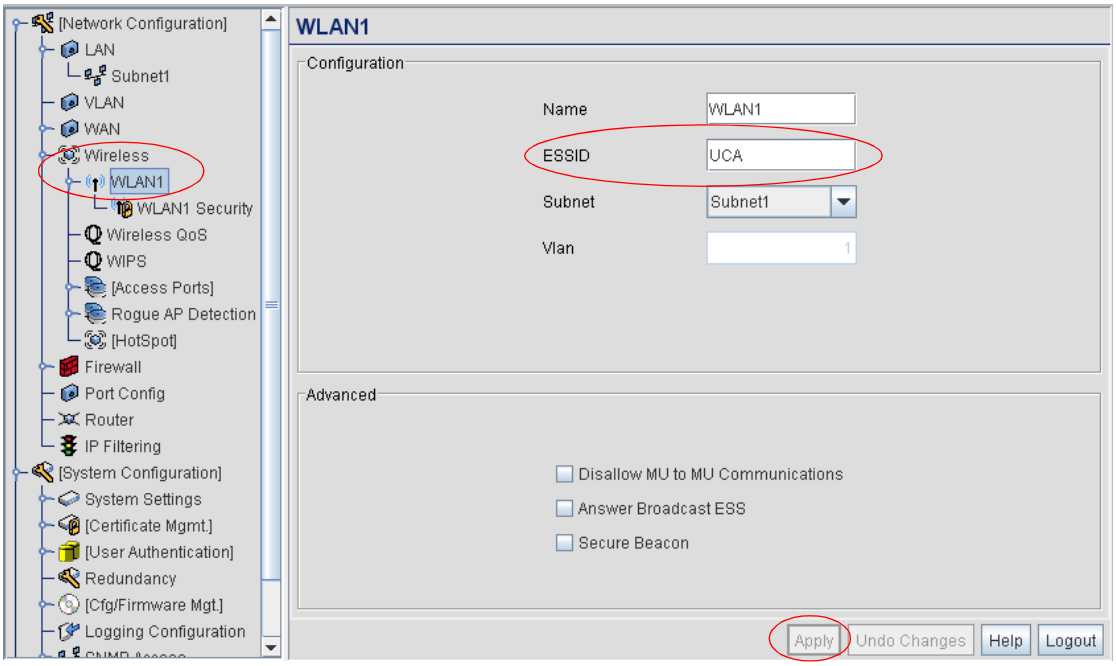


Figure E-4 *WLAN1 - ESSID*

- In the *ESSID* field, change *101* to *UCA*.
- Click **Apply**.

6. On the left side of the window under [Network Configuration], select *WLAN1 Security.* Ensure the radio buttons next to *No Authentication* and *No Encryption* are selected. Click **Apply** if required.

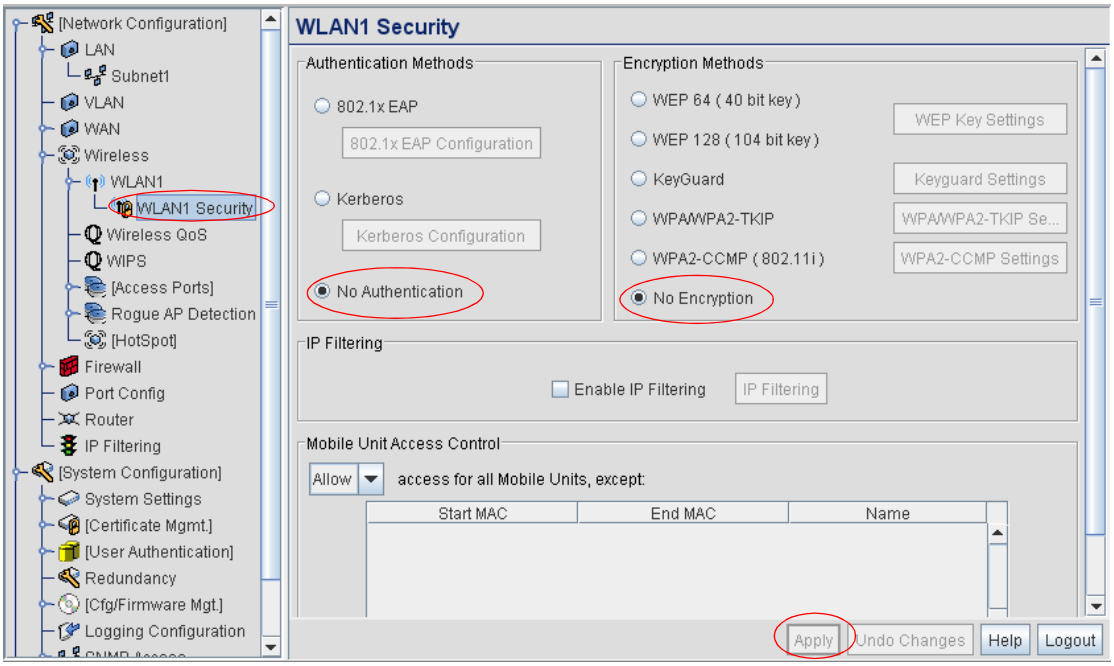


Figure E-5 *WLAN1 Security*

7. On the left side of the window under [Network Configuration], select *[Access Ports]* > *802.11 a/b/g Defaults* > *Beacon Settings* to set DTIM values of 2.

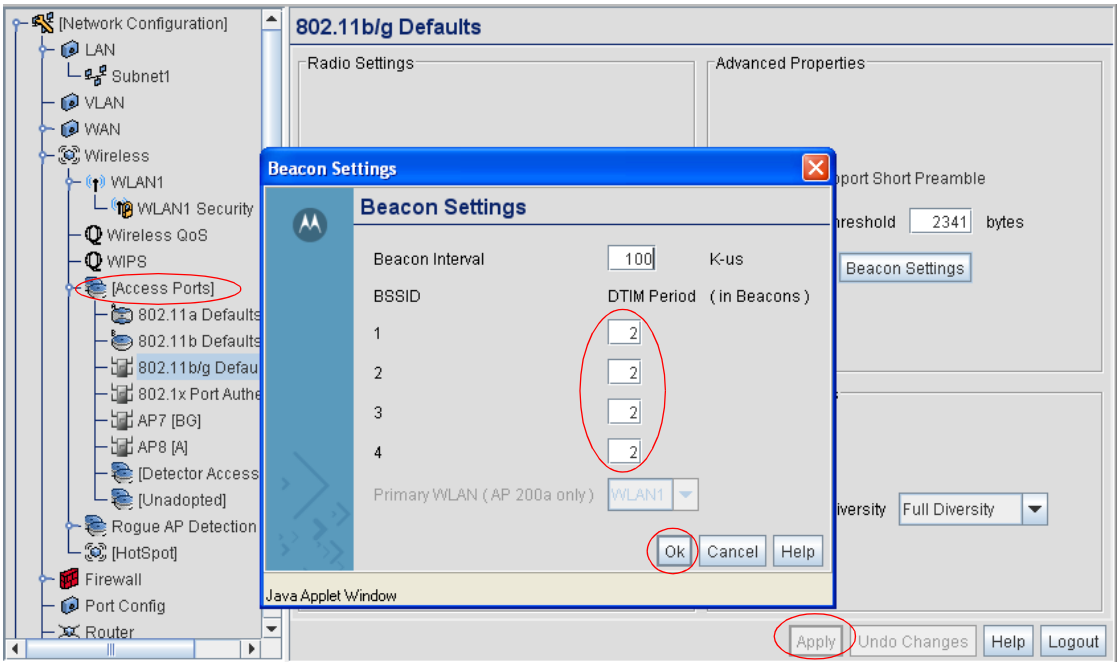


Figure E-6 *Beacon Settings*

- Change DTIM Per BSSID to 2.
- Click **OK**.
- Click **Apply**.

8. On the left side of the window under [Network Configuration], select *Wireless QoS*.

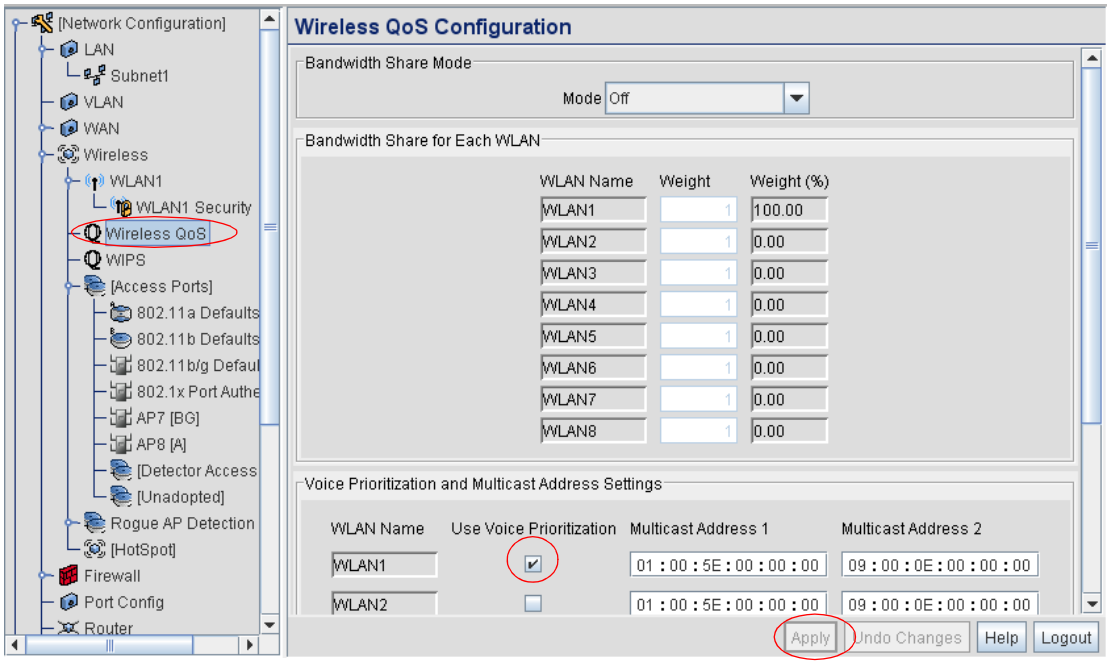


Figure E-7 *Wireless QoS Configuration*

- Check *Use Voice Prioritization* for WLAN1.
- Click **Apply**.

The RF hardware is now configured for the Demo Solution.

9. Close the browser.
10. Before continuing, validate:
    - the laptop's static IP address is set to 192.168.0.253
    - the subnet mask is set to 255.255.255.0.

    For details on how to check the computer IP address, see *page E-4*.
11. Ensure all firewalls are disabled on the laptop. For more details see *page 6-47*.

Step 6: Scan Rapid Deployment Bar Codes to Deploy CA50s

To deploy the CA50s onto the Demo Solution RF network and establish communication with the UCA server:

1. Insert the battery into the CA50. New units from the factory boot directly to the CA Menu.

   NOTE If the device does not boot to the CA Menu, see *Accessing the CA Menu on page 4-18*.

2. Launch the Rapid Deployment client from the CA Menu. For step-by-step instructions, see *Rapid Deployment on page F-1*.

3. Scan the 16 Rapid Deployment bar codes for the CA50 Demo Solution listed below. Scan these bar codes in the order they are listed.

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

4. When all bar codes have been scanned, the *"Your Device is Ready to Use"* message displays on the CA50. Press the right soft key to select **OK**.

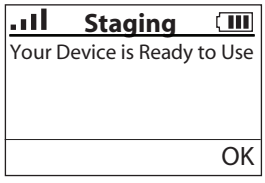


Figure E-8 *Staging Window*

5. From the CA Menu use the **Up** and **Down** keys to scroll and select *Warm Boot* to boot the CA50.

> ***NOTE*** To automate the warm boot of the device and automatically launch the UCA client application, see *page 12-2*.

6. The device CA50 is ready to use when the *Scan bar code to log in* displays.

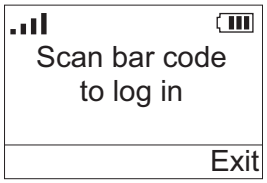


Figure E-9 *Login Prompt*

Step 7: Scan Employee Login Bar Code

Scan an employee (user) login bar code to log in to the UCA server. These bar codes correspond to a sampling of user data pre-loaded into the UCA server databases.

In some cases a PIN is required. If a PIN exists, it is shown below the login bar code. For information about employee identity authentication when entering PINs, see *Employee Identity Validation (Scanning a Login Bar Code and entering a PIN) on page 4-3*.

Managers

Bill Gordon - Managers
Manager (SM)
Ext 340
PIN Code: 98765

Sales Associates

Abigail Jones - Garden, Seasonal
Sales (Dept. Head)
Pin Code: None required

Anna Carter - Garden, Seasonal
Sales (New Hire)
Pin Code: None required

Kevin Smith - Bath, Electrical
Sales (Dept. Head)
Pin Code: None required

John Doe - Lumber, Bath
Sales
Pin Code: None required

Jane Doe - Electrical, Lumber
Sales
Pin Code: None required

Front End Associates

Joan Walker - Cashier
Front End
(Dept. Head)
Pin Code: None required

Dave Miles - Lot
Front End (New Hire)
Pin Code: None required

Step 8: Ready to Demonstrate the CA50 Voice and Scanning Functionality

Upon scanning the log in bar code, the CA50 is ready to be used to demonstrate any of the following functionality.

- **Employee Identity Validation**: Scanning Employee Login Bar Code and entering PIN - for details see *page 4-3*
- **Walkie-talkie communication** via the UCA client application - for details see *page 4-31*
- **Voice call communication** via the UCA client application - for details see *page 4-36*
- **Price lookup** via a business application - scan any SKU (below) - for details see *page 10-2*

Price Lookup SKUs for the Demo Solution

The SKUs below correspond to pre-populated retail items in the mini SKU database. This SKU database is provided for demo purposes and is not a part of or related to the UCA server software.

Scan the SKUs to perform a price lookup.

$4.50 - Power Strip 2pk

$0.79 - 1” Brush

$29.99 - Shower Massgr

$9.41 - GFCI Dplx Cvr

$69.00 - Dump Cart

Step 9. Modifying the Demo Solution: Creating New Users

To create a new user for the Demo Solution using the UCA server Web Console GUI, see *Step 6. Populate the UCA Server Database with 2 Users on page D-11*.

CAUTION Ensure to create unique external user IDs for new users to avoid duplicating users in the system.

In addition, bar codes must also be created for new users to login.

NOTE To delete users created in the Demo Solution, see *Delete Utility on page 9-52*.

Appendix F Rapid Deployment

Overview

The Rapid Deployment (RD) client on the CA50 client is used to scan staging bar codes. For details about the parameters contained within the RD bar codes, see *Staging and Deploying CA50s on page 12-2*.

NOTE 1. For a complete checklist outlining all the development and deployment activities required to implement a CA50 solution, see the *Development and Deployment Checklist on page 2-1*.

2. For a table listing the most commonly requested topics within this manual, see the Quick Startup Guide on the last page of the manual.

Accessing the Rapid Deployment Client

When the CA50 powers on for the first time, out-of-box, the CA Menu (see *CA Menu on page 3-12*). The default, highlighted selection on the CA Menu at the first power on is *Rapid Deployment* (see *Figure F-1*).

Scanning Rapid Deployment Bar Codes

To stage a CA50 over a wireless connection:

1. Ensure the UCA server setup is complete and ready to deploy data to the UCA client. (See *UCA Server Requirements and Setup on page 6-7*.)
2. Ensure the CA50 battery is fully charged (see *Charging Batteries on page 3-8*).
3. Power on the CA50 (see *Powering On the CA50 - Pre-deployment on page 3-11*)

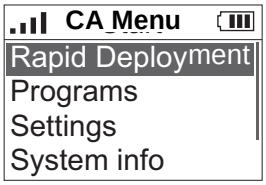


Figure F-1 *CA Menu Window*

4. To select and launch the *Rapid Deployment* option from the CA Menu, use the **Up** and **Down** keys to highlight the Rapid Deployment application; press the **Select** (**S**) key to launch the application.

When the Rapid Deployment application launches the *Scan* window displays.

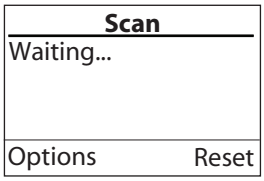


Figure F-2 *Scan Window*

5. When the *Scan* window displays *Waiting...* (as shown in *Figure F-2*), aim the laser at the appropriate bar code and press the **Scan** key to scan and decode an RD bar code (see *Scanning on page 4-16*). Upon a successful decode, a beep sounds and the CA50 LED blinks green. RD bar codes must be scanned in the appropriate order.
6. After scanning all the Rapid Deployment bar codes the *Processing* windows display the status, such as device registration or network connection.

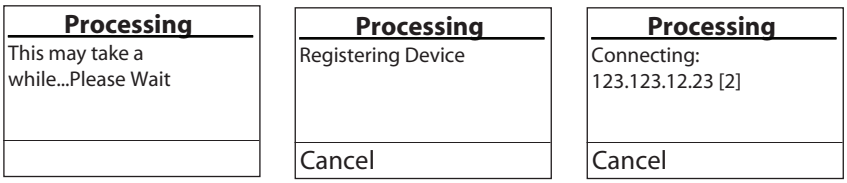


Figure F-3 *Processing Windows*

7. The deployment process continues to configure the device. When configuration of the device completes, the *Staging* window displays.

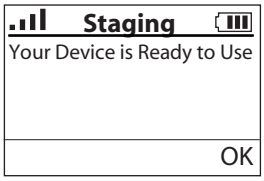


Figure F-4 *Staging Window*

8. Press the right soft key (*OK*) to continue.

When all deployment steps complete, a reboot may be required. If the device requires a reboot, the *Reboot Alert* window displays before the device boots.

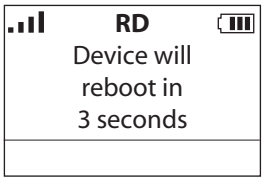


Figure F-5 *Reboot Alert Window*

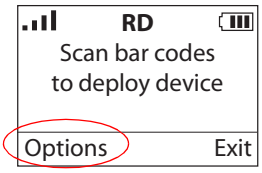
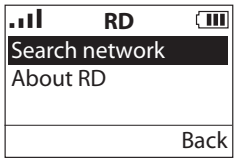
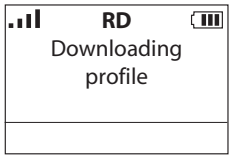
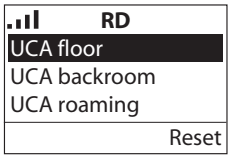
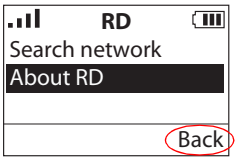
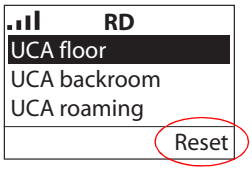
NOTE If the device required a reboot, processing on the device starts at the point it left off prior to the reboot.

When configuration is complete:

- The device has a valid network configuration and the CA50 is on the RF network.
- The UCA client application is ready to communicate with a UCA server.
- The settings are persistent.
- The device is ready for use.

Rapid Deployment Window Options

Table F-1 *Rapid Deployment Options*

Window Option	Description
Options 	Press the CA50 left soft key to display the default *Options* menu.  Scroll up and down the list and select an option: • Search Network - Searches the current adapter (wireless network) and performs actions based on the number of profiles established in the UCA server. For example, if only one store profile was defined in the UCA server, the network is searched, profile data downloads to the device and triggers the deployment process. If multiple department profiles were defined in the UCA server, the network is searched and a profile list displays. Select a profile from the list to download that department profile data to the device which triggers the deployment process.  Single Profile  Multiple Profiles • About RD - Displays information about Rapid Deployment.  Press the CA50 right soft key to go back to the previous window.
Reset 	Cancels an operation or window and return to the start window.
Exit	Closes the application and returns to the CA Menu. Note: Exiting the application prior to scanning all required bar codes discards any scanned data collected up to that point.

Errors

NOTE If the UCA client application cannot connect to the server, it continues to retry until you cancel (exit) the application. If failure to connect to the server persists, see the MSP Administrator.

(Page intentionally blank.)

Appendix G ActiveSync

Overview

To communicate with host computer devices install Microsoft ActiveSync on the host computer. Use ActiveSync to manually update data on the CA50 client and synchronize information on the client with information on the host computer. Changes made on the client or host computer appear in both places after synchronization.

ActiveSync software:

- Allows working with client-compatible host applications on the host computer. ActiveSync replicates data from the client so the host application can view, enter and modify data on the client.
- Synchronizes files between the client and host computer, converting the files to the correct format.
- Uploads and downloads data to the client from the host computer. Synchronization is a one-step procedure that ensures the data is always safe and up-to-date.
- Copies (rather than synchronizes) files between the client and host computer.
- Controls when synchronization occurs by selecting a synchronization mode, e.g., set to synchronize continually while the client is connected to the host computer, or set to only synchronize on command.
- Selects the types of information to synchronize and control how much data is synchronized.

NOTE The CA50 cannot be charged over an ActiveSync connection.

NOTE The CA50 supports a USB client connection only.

NOTE 1. For a complete checklist outlining all the development and deployment activities required to implement a CA50 solution, see the *Development and Deployment Checklist on page 2-1*.

2. For a table listing the most commonly requested topics within this manual, see the Quick Startup Guide on the last page of the manual.

Installing ActiveSync

If ActiveSync is not already installed on the host computer, download the latest version of the software from the Microsoft Web site at http://www.microsoft.com. Refer to the installation and RAS instructions included with the ActiveSync software.

NOTE Microsoft recommends installing ActiveSync on the host computer before connecting the client.

Setting Up an ActiveSync Connection

1. Select *Start* > *Programs* >*Microsoft ActiveSync* on the host computer. The *ActiveSync* window displays.

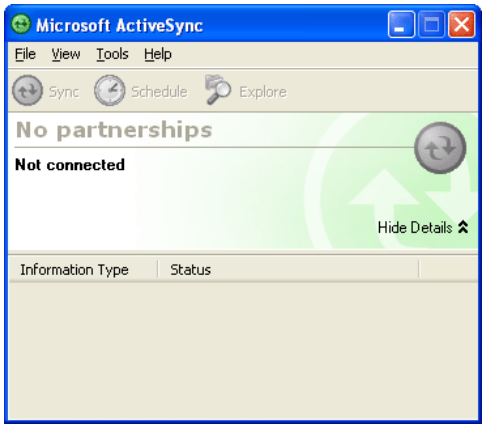


Figure G-1 *ActiveSync Window*

2. In the *ActiveSync* window, select *File* > *Connection Settings*. The *Connection Settings* window displays.

Figure G-2 *Connection Settings Window*

3. Ensure the *Allow USB connections* check box is selected.
4. Select the *Show status icon in taskbar* check box.
5. Select **OK** to save any changes made.

Connecting to the CA50

After ActiveSync installation is complete, the ActiveSync Setup Wizard may start automatically to set up a partnership between the CA50 and host computer. Cancel the Wizard set up as setting up a partnership is not required. When the CA50 is connected to the host computer, a connection initiates automatically and no setup is required.

To create a connection between the CA50 and host computer:

1. Connect the small end of the USB cable to the CA50 client and the large end of the USB cable to the appropriate USB client port on the host computer.
2. When the CA50 client device is detected by the host computer, the ActiveSync Connected window displays.

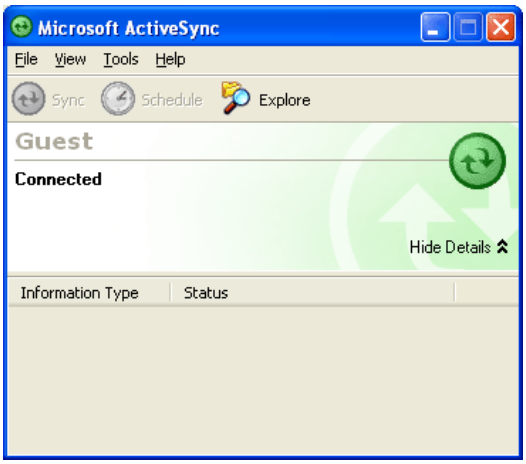


Figure G-3 *Connected Window*

Downloading Data using an ActiveSync Connection

To download data from the host computer to the CA client via ActiveSync:

1. Connect the CA50 client to the host computer using a USB cable.

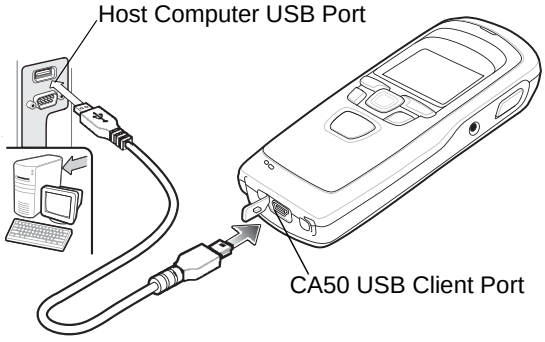


Figure G-4 *Connecting the CA50*

(Page intentionally blank.)

Appendix H UCA Server Backup and Restore

UCA Backup & Restore

The UCA server backup and restore process allows an administrator to restore the UCA server configuration in the event of a system failure. It also allows an administrator to duplicate the configuration of a UCA server in production to use as a quick setup for other servers.

Key Features

The UCA Backup & Restore has the following three features:

1. **UCA Backup** - Archives the configurations of the UCA database and UCA profiles (XML profile files).
2. **Restore** - Restores the backed up data on the target server and also the profiles. This action cleans up the existing contents of the database and restores data from the backup.
3. **Reconfigure** - Allows the UCA server administrator to reconfigure the Server IP Address, Store ID, PBX IP Address, PBX Port Number and cleanup the temporary tables, phone extension table and device users table.

Configuration Replication: To replicate configuration on other servers perform a complete restore (as described in Restore) of a backed up UCA database and reconfiguration. Configuration replication restores profiles and the UCA database. The administrator configured groups, roles and profiles on one UCA server can be replicated and imported to UCA servers in other locations.

Provide Database Credentials

For all three operations (Backup, Restore and Reconfigure) provide the details of the database server IP address and SA password.

- IP address: IP address of the database server. Since backup and restore runs on the same host as the UCA server, the default value of local host need not be changed.
- SA Password: Database System admin password provided during installation. See *Installing UCA Server Software on page 6-10* for password information.

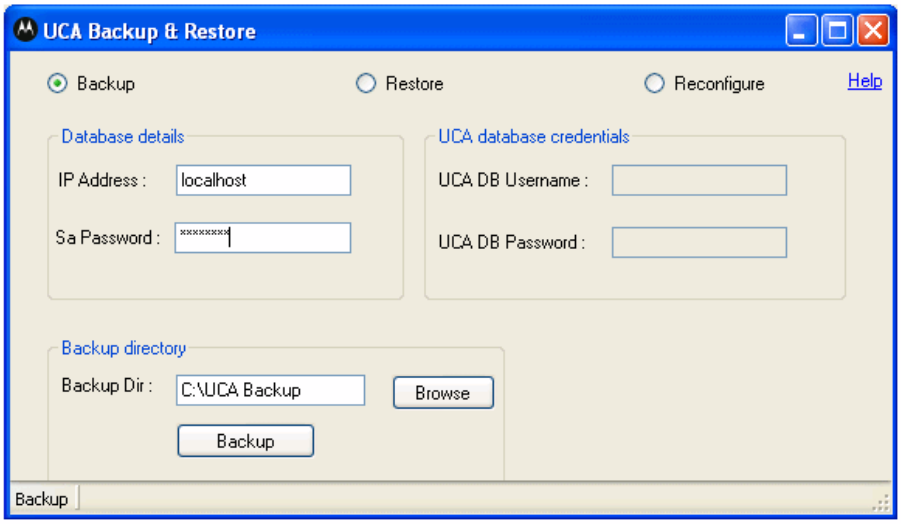


Figure H-1 *UCA Backup and Restore*

Backup the UCA Server

The backup archives the UCA database and profile files (profile files reside in: C:\Program Files\Motorola\UCA\Server\Profiles*.xml).

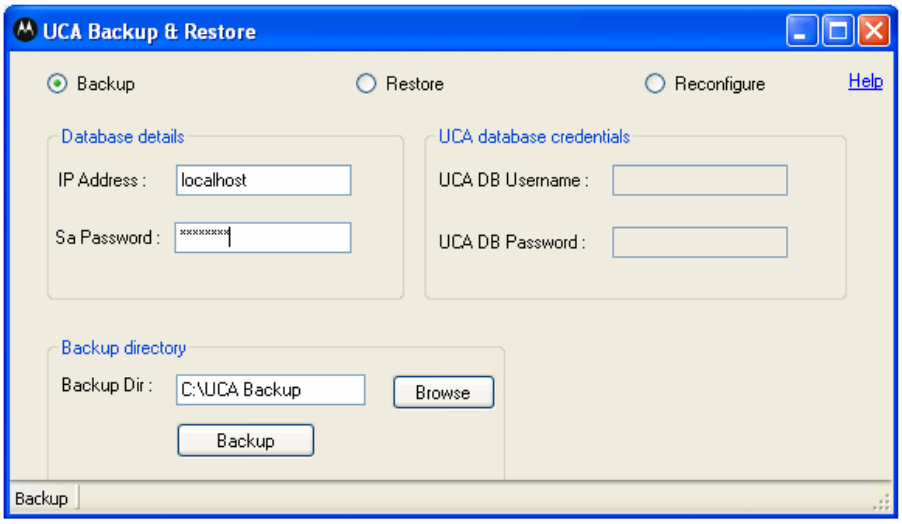


Figure H-2 *Back Up the UCA Server*

To backup the UCA server:

1. Provide the database credentials (see *Provide Database Credentials on page H-2*).
2. Click **Browse** and choose the directory location for the backup archive. The default location is C:\UCA Backup.
3. Click **Backup** to begin backing up the UCA database and profiles.
4. A message box appears and indicates that the backup operation was successful.

Restore the UCA Server

The Restore process restores the UCA database and the profiles from the backup (taken using the backup process).

NOTE If the version of the archived configuration of the UCA server does not match the target server, the application displays a message and shuts down.

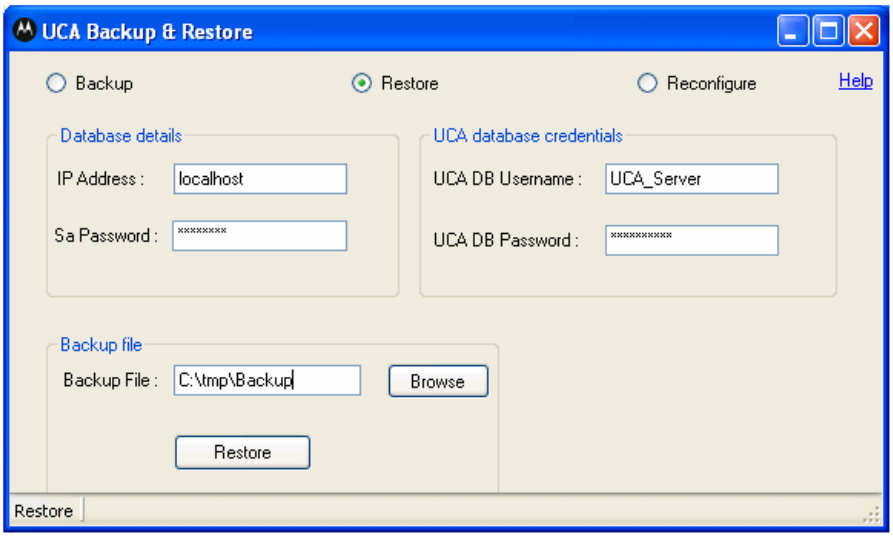


Figure H-3 *Restore the UCA Server*

To restore the UCA database and profiles:

1. Provide the database credentials sa login (see *Provide Database Credentials on page H-2*).
2. Provide the UCA Database login credentials provided during the installation. Refer to section 6-12 of CA50/UCA Client & Server Developer & User Guide.
3. Choose the backup archive by clicking **Browse**.
4. Click **Restore** to begin restoring the UCA database and profiles.
5. A message box appears and indicates that the restore operation was successful.

Reconfigure the UCA Server

The reconfiguration updates the UCA Server IP Address, Store ID, PBX IP Address, PBX Port Number, IIS Port Number and cleans the transient information present in database tables along with the UCA Users and Phone Extension table.

NOTE It is not necessary to supply all fields for reconfiguration. Reconfigure any one of the parameters or all the parameters.

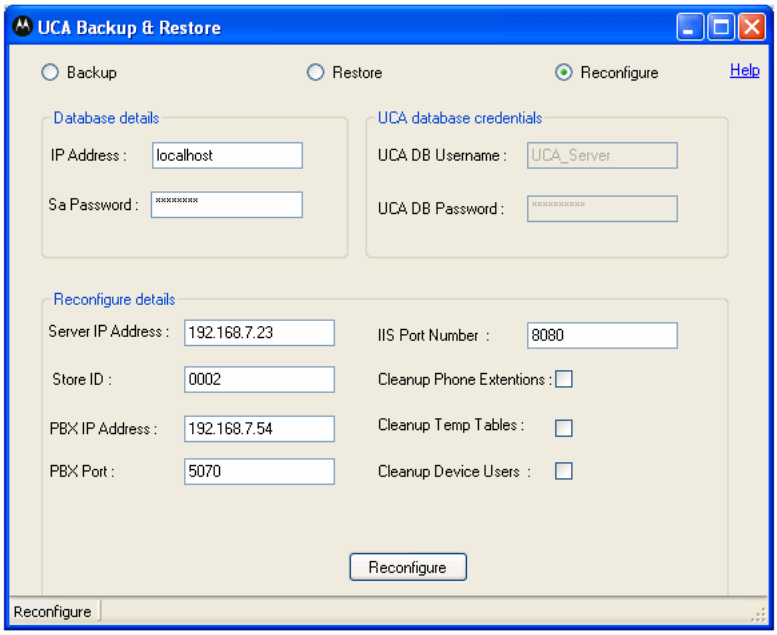


Figure H-4 *Reconfigure the UCA Server*

To reconfigure the UCA Server:

1. Select the **Reconfigure** radio button.
2. Server Information (Store ID, Server IP Address, PBX IP Address, PBX Port and IIS Port Number) and Cleanup fields are available for the reconfigure process.
3. Any of the specific information (server IP, PBX IP, etc.) can be configured separately.
4. Click **Reconfigure** to initiate the operation.
5. A message box indicates that the reconfigure operation is successful.

Appendix I Business Applications Supplement

Introduction

This chapter provides examples and tips for writing business applications for the CA50 device.

Adding a Menu Option

In this example, the developer can provide an option to exit the UCAClient on the device and return to the CA50 menu (shell). To accomplish this, the developer must add an additional menu in the profile.

To add a menu to the profile, use the following code structure:

```
<Menu ID="1" name="SalesMenu" title="Menu">
     <entry ID="1" value="Applications:M:L:Applications"/>
     <entry ID="2" value="WT Channels:M:L:SalesWTChannels"/>
     <entry ID="3" value="PhoneBook:M:L:PhoneBook"/>
     <entry ID="4" value="Parked calls:M:L:CallPick"/>
     <entry ID="5" value="Key lock:S:L:KeyLock"/>
     <entry ID="6" value="Do Not Disturb:S:L:QuietMode:Y"/>
     <entry ID="7" value="Settings:M:L:Settings"/>
     <entry ID="8" value="Log Off:S:L:LogOffConfirm"/>
     <entry ID="9" value="Power Off:S:L:PowerOff"/>
     <entry ID="10" value="About:S:L:About"/>
     <entry ID="11" value="Exit:S:L:ExitClient"/>


</Menu>
<Menu ID="2" name="SalesAppMenu" title="Options">
    <entry ID="1" value="WT Channels:M:L:SalesWTChannels"/>
    <entry ID="2" value="PhoneBook:M:L:PhoneBook"/>
    <entry ID="3" value="Parked calls:M:L:CallPick"/>
    <entry ID="4" value="Do Not Disturb:S:L:QuietMode:Y"/>
    <entry ID="5" value="Log Off:S:L:LogOffConfirm"/>
    <entry ID="6" value="Exit:S:L:ExitClient"/>
</Menu>
```

Launching a Business Application on UCAClient Startup

In this example, the developer can provide an option to launch a business application when the user logs into the UCAClient. No user action is needed. When the user logs in, the business application displays.

To allow the developer to customize the URL to launch, an entry is provided in the profile xml called "AutoLaunchHotLink" (see below). The attribute "URL" should contain the URL of the business application to launch.

```
<Desktop>
    <Link type="DefaultScan" URL="from DB"/>
    <Link type="UpHotLink" URL=""/>
    <Link type="DownHotLink" URL=""/>
    <Link type="FPTTHotLink" URL=""/>
    <Link type="SPTTHotLink" URL=""/>
    <Link type="AutoLaunchHotLink" URL=""/>
</Desktop>
```

XSD changes required (for developers):

```
<xs:complexType name="LinkType">
  <xs:attribute name="type" use="required">
    <xs:simpleType>
      <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="DefaultScan"/>
        <xs:enumeration value="DownHotLink"/>
        <xs:enumeration value="FPTTHotLink"/>
        <xs:enumeration value="SPTTHotLink"/>
        <xs:enumeration value="UpHotLink"/>
        <xs:enumeration value="AutoLaunchHotLink"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:attribute>
  <xs:attribute name="URL" type="xs:string" use="required"/>
</xs:complexType>
```

Setting the HTML Page Title

Each business application page can display its own title on the first line of the display on the CA50 device. In this example, the developer must include the following meta tag in the business application.

Meta tag:

```
<meta http-equiv="Title" content="Appl"/>
```

Appl displays in the first line.

IMPORTANT The length of *Title* is limited to a maximum of nine characters.

Adding a Timer to Visible and Vibrate Indicators

To set the timer for *Visible* and *Vibrate* indicators in a business application, use the following properties:

- Control the *On* duration for LED and Vibrate
- Control the *Off* duration for LED and Vibrate
- Control the number of cycles the two features run through.

By specifying the appropriate *On/Off* duration and cycle count, the timer can be set on these indicators.

The following meta tags are provided for setting the timer for *Visible* and *Vibrate* indicators.

```
<meta http-equiv="visibleindicator"
content="TYPE=GREEN;STATE=BLINK;ONDUR=100;OFFDUR=500;CYCLECNT=10;"/>
<meta http-equiv="vibrateindicator"
content="TYPE=VIBRATE;STATE=BLINK;ONDUR=500;OFFDUR=100;CYCLECNT=10;"/>
```

Table I-1 *Visible/Vibrate Indicator Attributes*

Attribute	Visible Indicator Allowed Values	Vibrate Indicator Allowed Values
TYPE	Red, Green or Amber	Vibrate
STATE	On, Off or Blink On - LED is always on Off - LED is always off Blink - LED changes state between on/off for the duration specified in ONDUR and OFFDUR	On, Off or Blink On - Vibrate is always on Off - Vibrate is always off Blink - Vibrate indicator changes state between on/off for the duration specified in ONDUR and OFFDUR
ONDUR	Time in milliseconds the LED is on.	Time in milliseconds the *Vibrate* is on.
OFFDUR	Time in milliseconds the LED is off.	Time in milliseconds the *Vibrate* is off.
CYCLECOUNT	Number of blink cycles.	Number of *Vibrate* cycles.
ONDUR, OFFDUR and CYCLECOUNT are application only if the STATE = BLINK		

Enabling the BackLight for HTML Pages

To enable a backlight on the CA50 device, the following meta tag must be added to the HTML page. By default, the backlight is disabled on all HTML pages.

```
<meta http-equiv="bklight" content="ON"/>
```

When the meta tag value is set *ON* the backlight lights when the page launches.

Download Files / Delete Downloaded Files

Download Files to the CA50 Device Upon User Login

The UCAClient has the capability to download files from an HTTP server to the CA50 device. This feature is useful for downloading custom .wav files to use with notification or business applications.

Information about the types of files to download can be provided in the profile xml as follows:

```
<DownloadFiles DownloadAtStartUp="Y">ank
    <File HTTPURL="" PathOnDevice="" Overwrite="N"/>
    <File HTTPURL="" PathOnDevice="" Overwrite="N"/>
    <File HTTPURL="" PathOnDevice="" Overwrite="N" />
</DownloadFiles>
```

Attributes

- DownloadAtStartUp="Y" — Enter "Y" to download the files soon after user login.
- HTTPURL — Source file location and name.
- PathOnDevice — Destination (path on the device); the default path on the CA50 device to which files are downloaded is "\Application\Download\".
- OverWrite="N" — Enter "Y" to overwrite the existing file on the device, if applicable.

Example 1

```
<File HTTPURL="http://server.somecompany.com/Response.wav" PathOnDevice="" Overwrite="N" />
```

Source = "http://server.somecompany.com/Response.wav"

Destination = "\Application\Download"

Example 2

```
<File HTTPURL="http://server.somecompany.com/Wav/Response.wav" PathOnDevice="\Wav\" Overwrite="N"
/>
```

Source = "http://server.somecompany.com/Wav/Response.wav"

Destination = "\Application\Download\Wav\"

Download Files to the CA50 Device After User Login

The UCAClient has the capability to download files from an HTTP server, to the CA50 device, at a point in time after user login. A menu option can be provided to allow the user to download files any time.

Example

```
<Menu ID="4" name="Applications" title="Applns">
  <entry ID="1" value="Download Files:S:L:DOWNLOADFILES"/>
  <entry ID="3" value="Price Check:S:S:http://server.somecompany.com/Sample.aspx"/>
</Menu>
```

The *Download Files* menu option displays on the *Application* menu. When selected, files are downloaded onto the CA50device.

Delete Downloaded Files

A menu option can be provided to allow the CA50 device user to delete all downloaded files in the folder "\Application\Download\".

Example

```
<Menu ID="4" name="Applications" title="Applns">
  <entry ID="1" value="Download Files:S:L:DOWNLOADFILES"/>
  <entry ID="2" value="Delete Files:S:L:DELETEALLDOWNLOADEDFILES"/>
  <entry ID="3" value="Price Check:S:S:http://server.somecompany.com/Sample.aspx"/>
</Menu>
```

The *Delete Files* menu option displays on the *Application* menu. When select, all files the '\Application\Download\' directory are deleted.

Adding User Information to the HTTP Header

User ID, User Group and User Role information can be added to the HTTP header. All HTTP requests originating from the UCAClient contain the information shown below in the HTTP header example. All information is available only after user login.

Example HTTP Header

GET /Motorola/UCA/Samples/UCASampleApp/Sample.aspx HTTP/1.1
Accept: */*
IPAddress: 192.168.1.138
SerialNo: 8106000
MAC: 0015709BF1CC
PlatformInfo: SYMBOL WinCE
UserID: 1100
UserGroup: Bath
UserRole: Sales
User-Agent: htmlayout 3.2; Windows CE; UcaClient)
Host: symbol-rvf2jb1t
Connection: Keep-Alive
Cookie: ASP.NET_SessionId=3c20tt45zw34nw45mcs01a45

Assigning a Default Value to the SpinBoxControl

To specify a default value for the SpinBoxControl displayed on the CA50 device, an additional attribute must be included in the HTML element for SpinBoxControl. (Also see *Sample HTML Code - Add Spin Boxes/Post Quantity to Server on page I-9*)

```
spinboxDefault=""
```

Example

```
<input type="submit" id="spinbox" name="x" SpinBoxLabel="xyz"
SpinBoxLabelPos="beforespinbox" ShowArrows="no" SpinBoxLine="2"
SpinBoxMaskDigit="?" value="??-??-??" spinboxDefault="22-07-09"
LeftBtnLbl="next#" RightBtnLbl="cancel"/>
```

Code Samples for HTML Pages in Business Applications

Enlarging Fonts in a Menu List

In this example, the following four items display in the CA50 Menu list:

- Work Stock
- Move Cages
- Manage Cages
- Mange Aisle

Given that the fonts are enlarged, two menu items appear at time. Pressing the **Down** arrow key moves down the list one item at a time.

Figure I-1 *Moving Down the Large Font Menu List*

Sample HTML Code

```
<body>
<form method ="get"
action="http://192.168.7.138/motorola/uca/samples/UCASampleApp/exitapplication.aspx" >

<select type="select" style="font:14px Verdana; width:88px; border:0px" name="menu" id="menu"
runat="server" size="3" autofocus>
            <option selected="selected" value="1">Work Stock</option>
            <option value="2">Move Cages</option>
            <option value="3">Manage Cage</option>
            <option value="4">Manage Aisle</option>
        </select>

<input name="LeftSoftKey" type="submit" id="RightSoftKey" value="Exit" />
</form>
</body>
```

Entering a Date in the Spin Box (HTML)

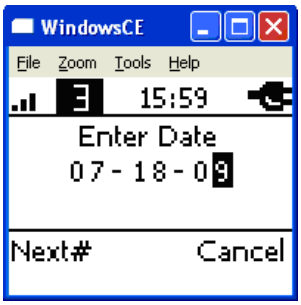


Figure I-2 *Date in Spin Box*

Sample HTML Code

```
<body>
<form name="form1" method="get"
action="http://192.168.7.138/motorola/uca/samples/UCASampleApp/exitapplication.aspx"
id="form1">

<input type="submit" id="spinbox" name="USERID" SpinBoxLabel="Enter Date"
    SpinBoxLabelPos="beforespinbox"  ShowArrows="no" SpinBoxLine="2"
SpinBoxMaskDigit ="?" value="??-??-??"LeftBtnLbl ="Next#" RightBtnLbl ="Cancel"/>
</form>
</body>
```

Scanning Item/Enter Quantity Using Spin Boxes

When scanning an item and entering the item's quantity, the first screen display directs the user to scan an item (for which the stock quantity will be entered).

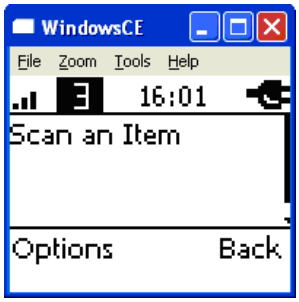


Figure I-3 *Scan an Item*

The next screen display requires the user to enter the stock quantity (see *Sample HTML Code - Add Spin Boxes/Post Quantity to Server* to use *Spin* boxes to enter the quantity).

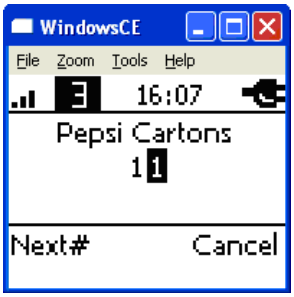


Figure I-4 *Enter a Quantity*

Sample HTML Code - Add Spin Boxes/Post Quantity to Server

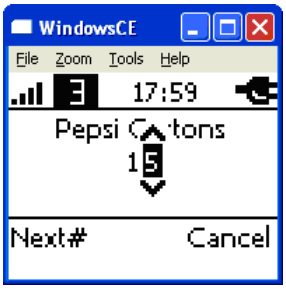


Figure I-5 *Enter a Quantity Using Spin Boxes*

```
<body>
<form method ="get"
action="http://192.168.7.138/motorola/uca/samples/UCASampleApp/exitapplication.aspx" >
<input type="submit" id="spinbox" name="USERID" SpinBoxLabel="Pepsi Cartons"
    SpinBoxLabelPos="beforespinbox"  ShowArrows="no" SpinBoxLine="2"
SpinBoxMaskDigit ="?" value="??"LeftBtnLbl ="Next#" RightBtnLbl ="Cancel"/>
</form>
</body>
```

The code below includes a default value for the *Spin* box.

```
<input type="submit" id="spinbox" name="USERID" SpinBoxLabel="Pepsi Cartons"
SpinBoxLabelPos="beforespinbox"  ShowArrows="no" SpinBoxLine="2" SpinBoxMaskDigit ="?"
value="??" SpinBoxDefault="15" LeftBtnLbl ="Next#" RightBtnLbl ="Cancel"/>
```

Adding and Enlarging Images/Fonts

Sample HTML Code - Enlarge Font Size

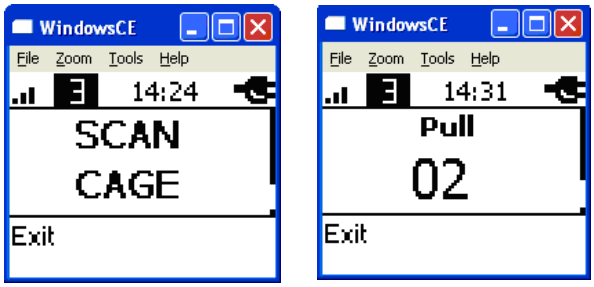


Figure I-6 *Enlarged Font*

Sample Code 1

```
<body>
<form name="form1" method="get"
action="http://192.168.7.138/motorola/uca/samples/UCASampleApp/exitapplication.aspx"
id="form1">

<text align="center" style="font:14px Verdana; width:88px; border:0px" ><b> SCAN </b></text>

<text align="center" style="font:14px Verdana; width:88px; border:0px" ><b> CAGE </b></text>

<input name="LeftSoftKey" type="submit" id="LeftSoftKey" value="Exit" />
<input name="RightSoftKey" type="button" id="RightSoftKey" value="" />

</form>
</body>
```

Sample Code 2

```
<body>
<form name="form1" method="get"
action="http://192.168.7.138/motorola/uca/samples/UCASampleApp/exitapplication.aspx"
id="form1">

<text align="center"><b> Pull </b></text>
<text align="center" style="font:20px Verdana; width:88px; border:0px" >02</text>

    <input name="LeftSoftKey" type="submit" id="LeftSoftKey" value="Exit" />
    <input name="RightSoftKey" type="button" id="RightSoftKey" value="" />
</form>
</body>
```

Sample HTML Code - Add Image

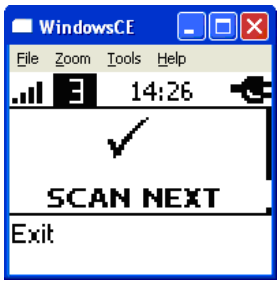


Figure I-7 *Add Image*

```
<body>
<form name="form1" method="get"
action="http://192.168.7.138/motorola/uca/samples/UCASampleApp/exitapplication.aspx" id="form1">

<text align="center"><img src="tick2.jpg">
    <br>
    <b>SCAN NEXT</b>
    </text>

<input name="LeftSoftKey" type="submit" id="LeftSoftKey" value="Exit" />
    <input name="RightSoftKey" type="button" id="RightSoftKey" value="" />

</form>
</body>
```

Sample HTML Code - Enlarge Image Size

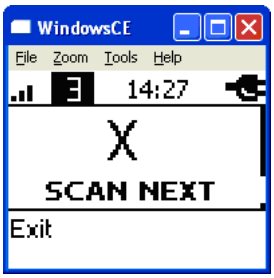


Figure I-8 *Enlarge Image*

```
<body>
 <form name="form1" method="get"
action="http://192.168.7.138/motorola/uca/samples/UCASampleApp/exitapplication.aspx" id="form1">

<text align="center" style="font:20px Verdana; width:88px; border:0px" > X </text>
    <text align="center"><b>SCAN NEXT</b></text>

<input name="LeftSoftKey" type="submit" id="LeftSoftKey" value="Exit" />
    <input name="RightSoftKey" type="button" id="RightSoftKey" value="" />

</form>
</body>
```

Enlarging Special Characters

Sample HTML Code - Enlarge Special Character Size

Figure I-9 *Enlarge Special Character*

```
<body>
<form name="form1" method="get"
action="http://192.168.7.138/motorola/uca/samples/UCASampleApp/exitapplication.aspx"
id="form1">

<text align="center">Ferraro Rocher</text>
<text align="center" style="font:20px Verdana; width:88px; border:0px" >&pound; 02</text>

    <input name="LeftSoftKey" type="submit" id="LeftSoftKey" value="Exit" />
    <input name="RightSoftKey" type="button" id="RightSoftKey" value="" />
</form>
</body>
```

NOTE The HTML code used for the CA50 device screen displays supports many special characters/symbols, such as the currency *pound* symbol shown in *Figure I-9*.

Note: Like pound, we can have other symbols as well.

LED Settings (HTML Business Application)

Sample HTML Tag

```
<meta http-equiv="visibleindicator" content="GreenFastBlink" />
```

LED Values

Table I-2 *Business Application LEDs*

Value	Description
RedSlowBlink	Red LED blinks once every second.
RedMedBlink	Red LED blinks once every 1/2 second.
RedFastBlink	Red LED blinks once every 1/10 second.
RedOn	Red LED remains on.
GreenSlowBlink	Green LED blinks once every second.
GreenMedBlink	Green LED blinks once every 1/2 second.
GreenFastBlink	Green LED blinks once every 1/10 second.
GreenOn	Green LED remains on.
AmberSlowBlink	Amber LED blinks once every second.
AmberMedBlink	Amber LED blinks once every 1/2 second.
AmberFastBlink	Amber LED blinks once every 1/10 second.
AmberOn	Amber LED remains on.

(Page intentionally blank.)

Appendix J Commands and Command Usage

Introduction

This chapter describes the commands available to developers and integrators which can be executed on the CA50 device. The chapter explains how the commands can be used in the profile, business applications and notification messages.

Commands Usage in Profiles

Profiles can be modified to customize menus displayed on the device. This section describes the commands and how to use them on CA50 menus.

Table J-1 *Profile Commands and Usage*

Command Name	Description	Usage in Profile As Part of Menus
SETSCANMODE	Set Walkie-Talkie into *Scan* mode. See *Device Modes on page 4-33* for details about *Do not disturb* mode.	SETSCANMODE Example: <entry ID="2" value="WT Scan Mode:S:L:SETSCANMODE"/>
SETCHANNEL	Set a particular Walkie-Talkie channel. See *Device Modes on page 4-33* for details about *Do not disturb* mode.	SetChannel:<Channel>:<Mode> Example: <entry ID="1" value="My WT Settings:S:L:SETCHANNEL:3:Normal"/>
SETRING	Set a ring type.	SetRing:<Wave File> Example: <entry ID="1" value="Basic Ring:S:L:SETRING:\Application\Wav\ringtone basic.wav"/>
PHNPLACECALL	Place a phone call to the given IP.	PHNPLACECALL:PBX(0) or P2P(1):extension:port:DTMF string Example: <entry ID="1" value="Call:S:L:PHNPLACECALL:1:192.168.7.172:5060:"/>
PHONEHANGUP	Hang up an active call.	PhoneHangUp Example: <entry ID="1" value="HangUp:S:L:PHONEHANGUP"/>
PHONEHOLD	Put an active call on hold.	PhoneHold Example: <entry ID="2" value="Hold:S:L:PHONEHOLD"/>
PHONERESUME	Resume a call on hold.	PhoneResume Example: <entry ID="3" value="UnHold:S:L:PHONERESUME"/>
PHNTRANSFERCALL	Transfer an active phone call to some other device/user.	PHNPTRANSFERCALL:PBX(0) or P2P(1):extension:port:DTMF string Example: <entry ID="1" value="172:S:L:PHNTRANSFERCALL:1:192.168.7.172:5060:"/>
PHONEMUTE	Mute an active call.	PhoneMute Example: <entry ID="9" value="PhoneMute:S:L:PHONEMUTE "/>

Table J-1 *Profile Commands and Usage (Continued)*

Command Name	Description	Usage in Profile As Part of Menus
QUIETMODE	Put the device in *Do not disturb* mode. See *Device Modes on page 4-33* for details about *Do not disturb* mode.	QuietMode:<Y/N> Example: <entry ID="5" value="Do Not Disturb:S:L:QUIETMODE:Y"/>
POWEROFF	Power off the device.	POWEROFF Example: <entry ID="5" value="Power Off:S:L:POWEROFF"/>
REBOOT	Reboot - Cold boot the device.	REBOOT Example: <entry ID="9" value="Reboot:S:L:REBOOT"/>
ABOUT	Display UCAClient *About* screen.	ABOUT Example: <entry ID="9" value="About:S:L:ABOUT"/>
SETBKLTLVL	Set backlight level.	SetBkltLvl:<Level 0-4> Example: <entry ID="1" value="low:S:L:SETBKLTLVL:0"/>
SETBKLTTIMEOUT	Set backlight timeout value.	SetBkltTimeout:<time in seconds> Example: <entry ID="1" value="2 secs:S:L:SETBKLTTIMEOUT:2"/> Value of 0 means the backlight is always on.
EXITCLIENT	Exit the UCAClient and go to the *Shell* menu.	ExitClient Example: <entry ID="11" value="Exit:S:L:EXITCLIENT"/>
DOWNLOADFILES	Initiate the download of files mentioned in the profile section <DownloadFiles>. See *User Profile Elements on page 7-4* more information.	DOWNLOADFILES Example: <entry ID="1" value="Download Files:S:L:DOWNLOADFILES"/>
DELETEALLDOWNLOADED FILES	Delete all files on the device in the download location \\Application\\Download. See *User Profile Elements on page 7-4* more information.	DELETEALLDOWNLOADEDFILES Example: <entry ID="2" value="Delete Files:S:L:DELETEALLDOWNLOADEDFILES"/>

Commands Usage in Business Applications (HTML Pages)

Commands can be used in HTML based business applications to perform various user based functions. For example, a user can make a phone call from the business application.

This section describes how to embed commands as part of business applications.

Table J-2 *Commands and HTML Usage*

Element	Description
<input type="Command" name="SETSCANMODE" options="" id="RightSoftKey" value="WTScan"/>	This input element allows business application developers to define a soft key for users to set the Walkie-Talkie in *Scan* mode. input type="Command" Command is a key word for the browser. It instructs the browser to execute the command specified in "name" when the specified key is pressed. name="SETSCANMODE" SETSCANMODE is a key word for the browser. On receiving this, the browser instructs the UCAclient to set the walkie talkie to Scan mode options="" This command does not take any parameters. id="RightSoftKey" Specifies the key that initiate the command. Valid values are "RightSoftKey" and "LeftSoftKey". value ="WTScan" Text to display over the soft key specified in "id". See *Device Modes on page 4-33* for details about WT channels and modes.

Table J-2 *Commands and HTML Usage (Continued)*

Element	Description
<input type="Command" name="SETCHANNEL" options="3:Normal" id="RightSoftKey" value="WTSetCh"/>	This input element allows business application developers to define a soft key for users to set the walkie-talkie channel. input type="Command" Command is a key word for the browser. It instructs the browser to execute the command specified in "name" when the specified key is pressed. name="SETCHANNEL" SETCHANNEL is a key word for the browser. On receiving this, the browser instructs the UCAclient to set the walkie talkie to the channel specified in the 'options' property options="3:Normal" Options is formatted as follows: Channel:Mode See *Device Modes on page 4-33* for details about WT channels and modes. id="RightSoftKey" Specifies the key that initiate the command. Valid values are "RightSoftKey" and "LeftSoftKey". value="WTSetCh" Text to display over the soft key specified in "id".
<input type="Command" name="SETRING" options="\Application\Wav\LASER.WAV" id="RightSoftKey" value="SetRing"/>	This input element allows business application developers to define a soft key for users to set the ring tone for incoming calls. input type="Command" Command is a key word for the browser. It instructs the browser to execute the command specified in "name" when the specified key is pressed. name="SETRING" SETRING is a key word for the browser. On receiving this, the browser instructs the UCAClient to set the ring tone on the device to the tone specified in the 'options' property options="\Application\Wav\LASER.WAV" Options is formatted as follows: "ToneFilename" This refers to the tone file that is present on the device. id="RightSoftKey" Specifies the key that initiate the command. Valid values are "RightSoftKey" and "LeftSoftKey". value="SetRing" Text to display over the soft key specified in "id".

Table J-2 *Commands and HTML Usage (Continued)*

Element	Description
<input type="Command" id="RightSoftKey" name="PHNPLACECALL" options="1:192.168.0.22:1720:012345" value="Expert"/>	This input element allows business application developers to define a soft key for users to make a phone call. input type="Command" Command is a key word for the browser. It instructs the browser to execute the command specified in "name" when the specified key is pressed. name="PHNPLACECALL" PHNPLACECALL is a key word for the browser. It instructs the browser to dial the string found in "options" when the specified key is pressed. options="1:192.168.0.22:1720:012345" Dialing string (DTMF supported format). Options is formatted as follows: PBX(0) or P2P(1):extension:port:DTMF string Where all :'s must be present even if it is PBX and there is no port or if there is no DTMF string Extension-IP address or an actual extension on the PBX Port - for P2P DTMF string. id="RightSoftKey" Specifies the key that initiate the command. Valid values are "RightSoftKey" and "LeftSoftKey". value="Expert" Text to display over the soft key specified in "id".
<input type="Command" name="PHONEHANGUP" options="" id="RightSoftKey" value="HangUp"/>	This input element allows business application developers to define a soft key for users to hang up when on an active phone call. input type="Command" Command is a key word for the browser. It instructs the browser to execute the command specified in "name" when the specified key is pressed. name="PHONEHANGUP" PHONEHANGUP is a key word for the browser. On receiving this, the browser instructs the UCAclient to hang up if on an active phone call options=" " This command does not take any parameter. id="RightSoftKey" Specifies the key that initiate the command. Valid values are "RightSoftKey" and "LeftSoftKey". value="HangUp" Text to display over the soft key specified in "id".

Table J-2 *Commands and HTML Usage (Continued)*

Element	Description
<input type="Command" name="PHONEHOLD" options="" id="RightSoftKey" value="Hold"/>	This input element allows business application developers to define a soft key for users to put an active call on hold. input type="Command" Command is a key word for the browser. It instructs the browser to execute the command specified in "name" when the specified key is pressed. name="PHONEHOLD" PHONEHOLD is a key word for the browser. On receiving this, the browser instructs the UCAclient to put the active call on Hold options=" " This command does not take any parameter. id="RightSoftKey" Specifies the key that initiate the command. Valid values are "RightSoftKey" and "LeftSoftKey". value="Hold" Text to display over the soft key specified in "id".
<input type="Command" name="PHONERESUME" options="" id="RightSoftKey" value="UnHold"/>	This input element allows business application developers to define a soft key for users to resume the call previously put on hold. input type="Command" Command is a key word for the browser. It instructs the browser to execute the command specified in "name" when the specified key is pressed. name="PHONERESUME" PHONERESUME is a key word for the browser. On receiving this, the browser instructs the UCAclient to resume the call previously put on 'Hold' options=" " This command does not take any parameter. id="RightSoftKey" Specifies the key that initiate the command. Valid values are "RightSoftKey" and "LeftSoftKey". value="UnHold" Text to display over the soft key specified in "id".

Table J-2 *Commands and HTML Usage (Continued)*

Element	Description
<input type="Command" id="RightSoftKey" name="PHNTRANSFERCALL" options="1:192.168.7.75:5060:" value="Transfer"/>	This input element allows business application developers to define a soft key for users to transfer an active call to another user. input type="Command" Command is a key word for the browser. It instructs the browser to execute the command specified in "name" when the specified key is pressed. name ="PHNTRANSFERCALL" PHNTRANSFERCALL is a key word for the browser. On receiving this, the browser instructs the UCAclient to transfer an active call to another user. options="1:192.168.0.22:1720:012345" Dialing string (DTMF supported format). Options is formatted as follows: PBX (0) or P2P(1):extension:port:DTMF string Where all :'s must be present even if it is PBX and there is no port or if there is no DTMF string Extension-IP address or an actual extension on the PBX Port - for P2P DTMF string. id="RightSoftKey" Specifies the key that initiate the command. Valid values are "RightSoftKey" and "LeftSoftKey". value="Transfer" Text to display over the soft key specified in "id".
<input type="Command" id="RightSoftKey" name="PHONEMUTE" options="" value="Mute"/>	This input element allows business application developers to define a soft key for users mute the line during an active call. input type="Command" Command is a key word for the browser. It instructs the browser to execute the command specified in "name" when the specified key is pressed. name="PHONEMUTE" PHONEMUTE is a key word for the browser. On receiving this, the browser instructs the UCAclient to 'Mute' during an active call options="" This command does not take any parameters. id="RightSoftKey" Specifies the key that initiate the command. Valid values are "RightSoftKey" and "LeftSoftKey". value="Mute" Text to display over the soft key specified in "id".

Table J-2 *Commands and HTML Usage (Continued)*

Element	Description
<input type="Command" id="RightSoftKey" name="QUIETMODE" options="" value="DND"/>	This input element allows business application developers to define a soft key for users to put the device in *Do not disturb* mode. See *Device Modes on page 4-33* for details about WT channels and modes. input type="Command" Command is a key word for the browser. It instructs the browser to execute the command specified in "name" when the specified key is pressed. name="QUIETMODE" QUIETMODE is a key word for the browser. On receiving this, the browser instructs the UCAclient to set the *Do not disturb* mode on. options="" This command does not take any parameters. id="RightSoftKey" Specifies the key that initiates the command. Valid values are "RightSoftKey" and "LeftSoftKey". value="DND" Text to display over the soft key specified in "id".
<input type="Command" name="POWEROFF" options="" id="RightSoftKey" value="PowerOff"/>	This input element allows business application developers to define a soft key for users to power off the device. input type="Command" Command is a key word for the browser. It instructs the browser to execute the command specified in "name" when the specified key is pressed. name="POWEROFF" POWEROFF is a key word for the browser. On receiving this, the browser instructs the UCAclient to power off the device options="" This command does not take any parameters. id="RightSoftKey" Specifies the key that initiates the command. Valid values are "RightSoftKey" and "LeftSoftKey". value="PowerOff" Text to display over the soft key specified in "id".

Table J-2 *Commands and HTML Usage (Continued)*

Element	Description
<input type="Command" id="RightSoftKey" name="REBOOT" options="" value="Reboot"/>	This input element allows business application developers to define a soft key for users to cold boot the device. input type="Command" Command is a key word for the browser. It instructs the browser to execute the command specified in "name" when the specified key is pressed. name="REBOOT" REBOOT is a key word for the browser. On receiving this, the browser instructs the UCAclient to cold boot the device options="" This command does not take any parameters. id="RightSoftKey" Specifies the key that initiates the command. Valid values are "RightSoftKey" and "LeftSoftKey". value="Reboot" Text to display over the soft key specified in "id".
<input type="Command" id="RightSoftKey" name="ABOUT" options="" value="About"/>	This input element allows business application developers to define a soft key for users to display the 'About' information for the UCAClient. It shows the UCAClient version information. input type="Command" Command is a key word for the browser. It instructs the browser to execute the command specified in "name" when the specified key is pressed. name="ABOUT" ABOUT is a key word for the browser. On receiving this, the browser instructs the UCAclient to display it's 'About' window options="" This command does not take any parameters. id="RightSoftKey" Specifies the key that initiates the command. Valid values are "RightSoftKey" and "LeftSoftKey". value="About" Text to display over the soft key specified in "id".

Table J-2 *Commands and HTML Usage (Continued)*

Element	Description
<input type="Command" id="RightSoftKey" name="SETBKLTLVL" options="4" value="Set Bklt"/>	This input element allows business application developers to define a soft key for users to set the backlight level on the device. Values range from levels 0-4. Values set this way remain set until they are manually reset or a cold boot is performed. input type="Command" Command is a key word for the browser. It instructs the browser to execute the command specified in "name" when the specified key is pressed. name="SETBKLTLVL" SETBKLTLVL is a key word for the browser. On receiving this, the browser instructs the UCAclient set the backlight level on the device to the value mentioned in the 'options' parameter. options="4" Options is formatted as follows: "BackLight Level" This value can range from 0-4. 0 is no backlight. 4 is for the brightest. id="RightSoftKey" Specifies the key that initiates the command. Valid values are "RightSoftKey" and "LeftSoftKey". value="Set Bklt" Text to display over the soft key specified in "id".
<input type="Command" id="RightSoftKey" name="SETBKLTTIMEOUT" options="1" value="Bklt Time"/>	This input element allows business application developers to define a soft key for users to set the backlight timer of the device. Value specified is in seconds. input type="Command" Command is a key word for the browser. It instructs the browser to execute the command specified in "name" when the specified key is pressed. name="SETBKLTTIMEOUT" SETBKLTTIMEOUT is a key word for the browser. On receiving this, the browser instructs the UCAclient to set the backlight timer on the device to the value mentioned in the 'options' parameter. options="4" Options is formatted as follows: "BackLight Timer in seconds" 0 is always 'ON'. id="RightSoftKey" Specifies the key that initiates the command. Valid values are "RightSoftKey" and "LeftSoftKey". value="Bklt Time" Text to display over the soft key specified in "id".

Table J-2 *Commands and HTML Usage (Continued)*

Element	Description
<input type="Command" id="RightSoftKey" name="EXITCLIENT" options="1" value="ExitUCA"/>	This input element allows business application developers to define a soft key for users to exit the UCAClient and go to *Shell* menu. input type="Command" Command is a key word for the browser. It instructs the browser to execute the command specified in "name" when the specified key is pressed. name="EXITCLIENT" EXITCLIENT is a key word for the browser. On receiving this, the browser instructs the UCAclient to exit itself. options="" This command does not take any parameters id="RightSoftKey" Specifies the key that initiates the command. Valid values are "RightSoftKey" and "LeftSoftKey". value="ExitUCA" Text to display over the soft key specified in "id".
<input type="Command" id="RightSoftKey" name="DOWNLOADFILES" options="" value="DwldFiles"/>	This input element allows business application developers to define a soft key for users to initiate a download of files mentioned in the profile xml associated with the user. See *User Profile Elements on page 7-4* more information. input type="Command" Command is a key word for the browser. It instructs the browser to execute the command specified in "name" when the specified key is pressed. name="DOWNLOADFILES" DOWNLOADFILES is a key word for the browser. On receiving this, the browser instructs the UCAclient to download files as mentioned in the profile xml associated with the user. options="" This command does not take any parameters id="RightSoftKey" Specifies the key that initiates the command. Valid values are "RightSoftKey" and "LeftSoftKey". value="DwldFiles" Text to display over the soft key specified in "id".

Table J-2 *Commands and HTML Usage (Continued)*

Element	Description
<input type="Command" id="RightSoftKey" name="DELETEALLDOWNLOADEDFILES" options="" value="DeleteFiles"/>	This input element allows business application developers to define a soft key for users to delete all the files downloaded previously to the folder \application\download\ on the device. See *User Profile Elements on page 7-4* more information. input type="Command" Command is a key word for the browser. It instructs the browser to execute the command specified in "name" when the specified key is pressed. name="DELETEALLDOWNLOADEDFILES" DELETEALLDOWNLOADEDFILES is a key word for the browser. On receiving this, the browser instructs the UCAclient to delete all the files downloaded previously to the folder \application\download\ on the device. options="" This command does not take any parameters id="RightSoftKey" Specifies the key that initiates the command. Valid values are "RightSoftKey" and "LeftSoftKey". value="DeleteFiles" Text to display over the soft key specified in "id".

Commands Usage in Notification Messages

The SendNotification API is used to notify the CA50 device user of any information in text format. This can also be extended to a complex work flow application. In this context, commands can be used in the notification to execute a command on the device. This section describes how commands can be embedded in *Notification* messages.

The SendNotification() Web service can be used by external programs to send an asynchronous message to a logged in CA50 device user. This Web service allows users to customize the left soft key and right soft key.

One of the parameters for the SendNotification(), NotifInfo, has the properties LSKey and RSKey which can be used to assign custom action.

Example:

notifInfo.LSKey = ""

notifInfo.RSKey = ""

See *Device Messaging Web Service on page 11-26* for details and instructions for using the Web service.

UCA Client commands can be assigned to these parameters.

Example:

notifInfo.LSKey = "WTScan:S:L:SETSCANMODE"

notifInfo.RSKey = ""

Using these parameters, the left soft key is *WTScan* and the right soft key is *Dismiss*.

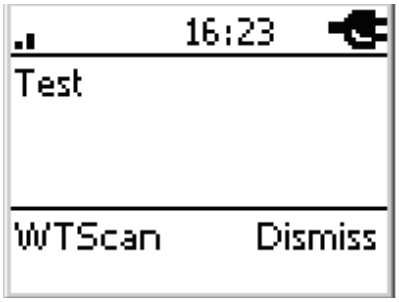


Figure J-1 *Assigned Left and Right Soft Keys*

NOTE When LSKey and RSKey are left blank, default values are assigned. The default value for LSKey is *Ack* and the default value for RSKey is *Dismiss*.

In the example in *Figure J-1*, pressing the left soft key sends the SETSCANMODE command to the UCAClient. The UCAClient then sets the WT mode to *Scan* mode.

Additional commands that can be used with SendNotification are listed in the table below.

Table J-3 *Commands Usage in Notification Messages*

Command Name	Description	Usage with Left and Right Soft Keys
SETSCANMODE	Set walkie-talkie into *Scan* mode. See *Device Modes on page 4-33* for more information about WT channels and modes.	WTScan:S:L:SETSCANMODE
SETCHANNEL	Set a particular walkie-talkie channel. See *Device Modes on page 4-33* for more information about WT channels and modes.	SetCh:S:L:SetChannel:3:Normal
SETRING	Set a ring type.	SetRing:S:L:SetRing:\Application\Wav\laser.wav
PHNPLACECALL	Place a phone call to the given IP.	Call:S:L:PHNPLACECALL:1:192.168.7.126:5060:
PHONEHANGUP	Hang up an active call.	HangUp:S:L:PHONEHANGUP
PHONEHOLD	Put an active call on hold.	Hold:S:L:PHONEHOLD
PHONERESUME	Resume a call on hold.	UnHold:S:L:PHONERESUME
PHNTRANSFERCALL	Transfer an active phone call to some other device/user.	Transfer:S:L:PHNTRANSFERCALL:1:192.168.7.75:5060:
PHONEMUTE	Mute an active call.	Mute:S:L:PHONEMUTE
QUIETMODE	Put the device on *Do not disturb* mode. See *Device Modes on page 4-33* for details about WT channels and modes.	SetDND:S:L:QUIETMODE:Y
POWEROFF	Power off the device.	PowerOff:S:L:POWEROFF
REBOOT	Reboot - Cold boot the device.	Reboot:S:L:REBOOT
ABOUT	Display UCAClient *About* screen.	About:S:L:About
SETBKLTLVL	Set backlight level	SetBkLT:S:L:SetBkltLvl:0
SETBKLTTIMEOUT	Set backlight timeout value.	BkLTTime:S:L:SetBkltTimeout:2 Note: 0 means backlight is always on.

Table J-3 *Commands Usage in Notification Messages (Continued)*

Command Name	Description	Usage with Left and Right Soft Keys
EXITCLIENT	Exit the UCAClient and go to the *Shell* Menu.	Exit:S:L:ExitClient
DOWNLOADFILES	Initiate download of files mentioned in the profile section <DownloadFiles>. See *User Profile Elements on page 7-4* for more information.	DwndFls:S:L:DOWNLOADFILES
DELETEALLDOWNLOADEDFILES	Delete all files on the device in the download location \\Application\\Download. See *User Profile Elements on page 7-4* for more information.	DelFiles:S:L:DELETEALLDOWNLOADEDFILES

Index

Numerics

1D bar codes 4-15
802.11 . . . . . 4-29, D-16

A

accessories
- cables B-12
- earbud B-2
- eight-slot battery charger B-2, B-8
- eight-slot CA50 charging cradle B-2, B-3
- lanyard B-2
- mounting brackets B-13, B-15
- server software B-2
- single-slot CA50 charging cradle B-5
- single-slot cradle B-2
- soft pouch B-2
- spare battery kit B-2
- specifications A-4
- wall mount bracket B-2

ActiveSync
- communication between CA50 and host G-3
- installing G-2
- setting up a connection G-2

airbeam 4-27
alerts profile elements 7-9
APIs 11-2
- console users 11-36
- device users 11-29
- groups 11-18
- phone extensions 11-14
- roles 11-22

application deployment 11-1, 12-1
- rapid deployment 4-19, F-1

application keys 3-2, 3-3, 4-7
audio indications 4-11
audio jack 3-3, 4-9
audio profile elements 7-4
audio test app 4-20, 4-21
authentication
- methods
  - authenticatemac 11-11
  - authenticateuser 11-12
  - authenticateuserpassword 11-13
  - getuserdata 11-8, 11-34
- web service 11-8

auto launch UCA client app 12-6

B

B_ConsoleUser table 11-37
B_Groups table . . . . . 11-10, 11-15, 11-18, ... 11-19, 11-20, 11-31
B_PhoneExt table 11-16
B_Roles table . . . . . 11-9, 11-22, 11-23, ... 11-24, 11-30, 11-37
B_SystemSettings table 11-6
B_User table . . . . . 11-11, 11-29, 11-32, ... 11-33, 11-34, 11-36
backlight settings 4-27
bandwidth
- decrease roam times through channel mask . . 13-3
- number of CA5os deployed per site 13-2
- wireless network optimization 13-2

bar code
- enable/disable C-1
- enabled C-2
- scan sample 4-19
- scanning 4-9, 4-16
- scanning functionality 1-2
- supported C-2

bar code login 4-3
bar codes

one dimensional 4-15
battery
accessories B-2
booting 4-8, 4-17
charging B-13, B-15
battery in CA50 3-8
eight-slot CA50 charging cradle B-2, B-3
eight-slot charger B-2, B-8
first time use 3-8
indications 4-10, 4-12, B-4, B-7, B-9
preparing CA50 for use 3-4
single slot cradle B-6
single-slot CA50 charging cradle B-5
single-slot cradle B-2
spare battery 3-4, 3-8, B-6, B-8
temperature 3-7, 14-6, A-2, A-4, A-5
time 3-7
inserting 3-4
installing a charged battery 3-11, 4-2
lithium-ion 3-1
low battery indicator, beep 4-11, 7-8
removing 3-5
spare battery kit B-2
spec A-1
strength 4-6
system info 4-30
system settings 6-48
troubleshooting 14-6, 14-7
battery charging
temperature A-4
battery management 13-6
device not in use 3-9
optimizing user device settings 3-9
out-of-box 3-9
wireless profile creation 3-9
beep
low battery indicator, beep 4-11
booting the device . . . . . 3-8, 3-11, 4-4, 4-8, ... 4-17, 10-7, 12-6, 14-6, F-2
bullets xx
business applications . . . . . 1-2, 4-54, 6-5, ... 10-2, 13-9, E-1
definition 10-2
employee identity validation 4-3, D-15, E-14
flashing LEDs 10-7
launching 10-3
playing audio 11-2, 11-44
sample code
add menu option I-2
add timer to indicators I-4
add user info to HTTP header I-6
assign default value to spin box I-7
download/delete files I-5
enable backlight I-5
launch app on startup I-3
set HTML page title I-4
sending text and audio 11-26
troubleshooting scanner 14-11
using the scanner 10-6, 10-8
vibrate device 10-7

C

CA menu
access 4-9
accessing when app loaded 4-18
at boot up 4-4
cold boot 4-17
configuration using rapid deployment F-3
definition 3-12
display 3-11
options 4-19
redeployment 4-2
sleep mode 4-8
sleep mode selection 3-13
text configuration using rapid deployment F-1
warm boot 4-8, 4-17
cable B-12
cables
communication setup B-12
call park 4-42, 4-47
call retrieve 4-50
calls 4-8
answering a voice call 4-45
initiating walkie-talkie activity 4-34
making a voice call 4-45
parking a voice call 4-47
PBX 4-37
receiving walkie-talkie communication 4-34
retrieving parked voice call 4-50
standard 4-37
transferring a voice call 4-46
walkie-talkie 4-31
certificate settings 4-29
channel modes 4-33, 7-5
channel settings, walkie-talkie 4-32, 7-4
channels, walkie-talkie 4-33
charge contacts 4-9
charging
not via activesync G-1
single slot cradle B-6
temperature A-2, A-4, A-5
charging battery . . . . . 3-4, 3-7, 3-8, ... 4-10, 4-12, 14-6, ... B-2, B-3, B-4, ... B-5, B-7, B-8, B-9
charging battery in device 3-8
charging indications 4-10, 4-12, B-4, B-7, B-9

charging temperature 3-7, 14-6, A-2, A-4, A-5
cleaning/disinfecting the CA50 14-3
code types C-2
cold boot 3-8, 4-8, 4-17, 10-7, 12-6
commands and HTML usage J-4
commands and usage
profiles J-2
communication
LEDs 3-2, 4-9
microphone 3-2, 4-9
receiver 4-9
scanning 3-2, 4-9, 4-15
speaker 3-2, 4-9
communication setup
communication/charge cables B-12
communication status 4-5
communication/charge cables
communication setup B-12
configuration, walkie-talkie 4-31
connection loss, CA50 and PBX 4-38
connectivity
initial 12-1
PBX parameters 7-9
RF B-4, B-7
connectors
activesync 4-9
earbud 3-3, 4-9
console user auth web service
methods
authenticateuserpassword 11-41
getuserdata 11-40
console users
creating 9-44
deleting 9-47
editing 9-45
editing user passwords 9-46
overview 9-43
conventions
notational xx
cradles
eight-slot battery charger B-2, B-8
eight-slot CA50 charging B-2, B-3
four slot spare battery charger
setup B-8
mounting brackets B-13, B-15
single slot USB serial B-6
single-slot B-2
single-slot CA50 charger B-5
creating additional extensions 9-15
creating new users 9-20
creating phone extension 9-9

D

data capture
laser scanning 4-15
one dimensional bar codes 4-15
scan angle 4-15
scan range 4-15
scan window 3-2, 4-9, 4-15
scan/select key 4-9
scanning 4-16
database 11-32, 11-34, 11-36
database tables
B_ConsoleUser 11-37
B_Groups 11-10, 11-15, 11-18, 11-19, 11-20, 11-31
B_PhoneExt 11-16
B_Roles 11-9, 11-22, 11-23, 11-24, 11-30, 11-37
B_SystemSettings 11-6
B_User 11-11, 11-29, 11-32, 11-33, 11-34, 11-36
B_User table 11-5
S_PhoneExtType 11-15
S_SystemVariables 11-6
S_UserStatusType 11-9, 11-30, 11-37
T_DebugTrace 6-49, 6-50, 6-51
date/time settings 4-29
debuglog profile elements 7-12
delete
entries in UCA database 9-52
UCA database 6-42
utility 9-52
delete utility, server GUI 9-52, 9-53
demo solution 6-2, 6-5
configure RF network E-5
employee login bar code E-14
overview E-1, E-2, E-17
RD bar codes E-11
required hardware E-2
SKU bar codes E-17
deployment 11-1, 12-1
deployment example
create XML profile D-6
deploy CA50s D-18
generate employee login bar code D-15
generate RD bar codes D-16
overview D-1, D-2
populate server with 2 users D-11
required settings D-3
deployment parameters 12-2
desktop profile elements 7-7
development toolkit 6-5
device config profile elements 7-7
device maintenance 14-3

device on 4-8
device user accounts
all users 9-19
creating new users 9-20
deleting users 9-24
edit users 9-21
editing user passwords 9-23
import files 9-25
recent users 9-18
disinfecting the CA50 14-3
display 3-2
display icons
battery strength 4-6
communication 4-5
sample screen 4-5
signal strength 4-5
title 4-5
down arrow key 3-2, 4-7
downloadfiles profile elements 7-6

E

earbud 3-3, 4-9, 4-13, B-2
eight-slot battery charger B-2, B-8
eight-slot CA50 charging cradle B-2, B-3
eight-slot spare battery charger wall mount . . B-13, B-15
employee login
generation of employee login bar code D-15
identity validation 4-3, 10-3, D-15, E-14
login 4-3
extensions porting 9-13

F

features
CA5070-0U0LF5KV11R xviii
CA5070-0U0LF5KV18R xviii
find utility 9-52
find utility, server GUI 9-52
firewalls, configuring 6-47
firmware *See operating system*
four slot spare battery charger
setup B-8
Fusion log 4-29

G

GUI 6-5

H

HTML usage J-4

I

IIS 6-8, 6-54
import utility 9-55
indications
audio 4-11
charging 4-10, 4-12
screen 4-12
information, service xxi
ingress protection rating 14-3
inserting battery 3-4
install
demo version UCA server software 6-20
production version UCA server software 6-11
UCA server software 6-10
installation software 6-8
Internet Information Server 6-54
Internet Information Services 6-8
IP PBX *See PBX*
IP rating 14-3

K

key attributes 7-23
key check on CA50 device 4-24
key on CA50
volume 3-3
keys on CA50
device on 4-8
down arrow 3-2, 4-7
front application 3-2, 4-7
mute 3-3, 4-7
mute microphone 4-8
mute speaker 4-9
quiet mode 3-3
scan/select 3-2, 4-7, 4-9
side application 3-3, 4-7
soft 3-2, 4-7
unicast calls 4-8
unmute microphone 4-8
unmute speaker 4-9
up arrow 3-2, 4-7
voice calls 4-8
volume 4-7, 4-9
walkie-talkie 3-2, 3-3, 4-7, 4-8

L

lanyard B-2
laser scanning *See data capture*
launching business applications 10-3
LED test on CA50 device 4-25, 4-26
LEDs 3-2, 4-9, 4-10
decode 4-16

listen only channel, walkie-talkie 4-32, 7-5
listen/talk channel, walkie-talkie 4-32, 7-5
lithium-ion battery 3-1
localization
staging 12-6
log file 7-12, 14-8
CA50 registry 14-8
disable 14-10
example 14-10
settings
agentdebuglevel 14-9
browserdebuglevel 14-9
logfilelocation 14-8
loggingenabled 14-8
override 14-10
phonedebuglevel 14-8
scannerdebuglevel 14-8
WTTdebuglevel 14-8
UCAClient.txt 14-8
XML profile 14-8
login
post-deployment 4-2
pre-deployment 3-11
login bar code
employee identity validation 4-2, 4-3, E-14
generating D-15
scanning 4-2, 4-3, E-14
login to server 9-3
lost RF connection 4-12

M

maintenance
cleaning/disinfecting the CA50 14-3
scheduled 14-2
UCA server 14-2
menu
CA *See CA menu*
states, code sample 7-22
menu options
programs 4-19
rapid deployment 4-19
menustates profile elements 7-12
messaging 1-2, 1-3, 4-55,
. . . . . .6-5, 6-9, 6-58,
. . . . . 11-26, 11-44, 11-47
methods
adddeviceuser 11-29
addextension 11-14
authenticatemac 11-11
authenticateuser 11-12
authenticateuserpassword 11-13, 11-41
getextensioninfo 11-17
getuserdata 11-8, 11-34, 11-40
phone book web service 7-19
removedeviceuser 11-32
removeextension 11-15
setdeviceuserpin 11-34
updatedeviceuser 11-33
updateextension 11-16
metricslog profile elements 7-11
microphone 4-8
mute 4-8
walkie-talkie 3-2, 4-9
mounting brackets B-13, B-15
MSP
license 12-1
provision 12-1
rapid deployment bar codes 12-2
generate D-16
scan F-1
server function 3-4
software 1-4, 12-1
staging 6-53, 9-47, 12-1
staging & localization 12-6
troubleshooting 14-4
warranty iii
multi-slot battery charger B-2, B-8
multi-slot charger B-2, B-3
mute key
quiet mode key 3-3

N

network connectivity
generate wireless profile 3-9
ping 4-27
RF network optimization 4-31
signal strength 4-22, 4-23
signal strength recommendation 3-10
wireless security protocols 4-14
network settings 4-29
no phone error 4-39
non-listen accessible channel, walkie-talkie . . 4-32, 7-5
notification messages, customize J-14

O

operating environment A-1
operating system
RD client F-1
upgrade 12-1
version 4-2, 4-29
optimize
battery management 13-6
implementation 13-2
scanner performance 4-15
user device settings 3-9

wireless infrastructure 3-10
wireless network for VoIP 3-10
optional APIs 11-2
console users 11-36
device users 11-29
groups 11-18
phone extensions 11-14
roles 11-22
OS upgrade *See operating system*
out-of-box
battery management 3-9
CA menu 3-12
CA menu options 4-19
CA50 client 5-1
first time use 3-11
license 12-1
power on F-1

P

paging . . . . . .1-2, 1-3, 4-55, ... 6-58, 11-26, 11-44, 13-7
park call
no available extension 4-49
on any available extension 4-47
on any available extension within group/department 4-48
troubleshooting 4-52
passwords
for server security 6-57
for validating server install 6-45
installing UCA server software 6-13, 6-21, 6-34
server login 6-45
server security 6-54
strong 6-13, 6-21, 6-22, 6-34, 6-35
strong definition 6-13, 6-21, 6-34
system administrator 6-13, 6-21, 6-34
PBX
and voice call 1-3, 6-2, 6-5
call park 4-47
calls 4-37
configuring an IP PBX 4-36, 4-38
importing and creating extensions 9-13, 11-14
integration 1-2, 4-36, 13-2, 13-8, 13-10
optional CA50 components 1-4
password 9-10
phone extension status 9-7
phone settings 6-48, 9-41
registration 7-4, 14-14
supported IP PBX protocols 4-36
system data 14-3
system phone book entries 9-14
UCA phone extension 11-47
PBX connectivity parameters 7-9
phone extension web service 11-14
bulk loading extensions 9-13
methods
addextension 11-14
getextensioninfo 11-17
removeextension 11-15
updateextension 11-16
phone profile elements 7-4
phone/walkie-talkie
additional phone entries 9-14
creating additional extensions 9-15
creating phone extension 9-9
deleting phone extensions 9-17
editing additional extensions 9-16
editing phone extension properties 9-11
phone extension imports 9-13
phone extension pool 9-8
phone extension status 9-7
PIN entry 4-3, 10-3, E-14
ping test on CA50 device 4-27
post-deployment
login 4-2
power on 4-2
pouch B-2
power off 3-13
power off cradle B-7
power on 4-8
post-deployment 4-2
pre-deployment 3-11
pre-deployment *See out-of-box*
profile builder
alert settings 8-16
client settings 8-15
create menus 8-21
create new profile 8-4
desktop settings 8-11
device settings 8-14
download files settings 8-10
metrics & logging 8-13
phone settings 8-6
volume settings 8-12
walkie-talkie settings 8-7
profile commands and usage J-2
profile management
creating an associated base profile 9-27
deleting an associated base profile 9-29
editing an associated base profile 9-28
profile roaming 4-29
profiles
base XML sample 7-25
completed user XML sample 7-27
elements
alerts 7-9
audio 7-4

debuglog 7-12
desktop 7-7
device config 7-7
downloadfiles 7-6
menustates 7-12
metricslog 7-11
phone 7-4
user 7-4
walkie-talkie 7-4
ring tone 7-16
programs
airbeam 4-27
audio test 4-20, 4-21
scansamp app 4-19, 14-11
settings 4-27
system info 4-29
UCA client app 4-19
UCA key check 4-24
UCA LED test 4-25, 4-26
UCA sys info 4-22, 4-23
UCAPing 4-27

R

rapid deployment 4-19, F-1
RD *See rapid deployment*
rebooting
cold boot 4-17
options 3-12
re-accessing the CA menu 4-18
warm boot 4-17
receiver 3-2, 4-9
removing battery 3-5
retrieve call
from any extension 4-50
from any extension within group/department . . 4-51
troubleshooting 4-52
RF network
activity 3-10
connectivity B-4
infrastructure 12-1
lost connection 4-12
maximizing run time 3-10
optimization for walkie-talkie 4-31
settings 3-4
signal 7-8
RF network connectivity B-7
ring tone 7-16
roles and groups
groups
creating 9-31
deleting 9-33
editing 9-32
overview 9-30
roles
creating 9-34
deleting 9-36
editing 9-35
overview 9-33

S

S_PhoneExtType table 11-15
S_SystemVariables table 11-6
S_UserStatusType table 11-9, 11-30, 11-37
scan/select key 3-2, 4-7
scanner
business app 4-54
business applications 1-2
business apps 10-2
client.desktop.link.defaultscan 6-48, 6-56
considerations 4-15
default scan 7-7
demo solution E-14
deviceconfig element 7-8
enabled code types C-2
engine C-1
indications 4-10, 4-11, 4-12
login 4-2, 4-3, 6-4
logoff 4-4
max power save 3-9
mode 4-5, 4-31, 4-33, 4-35, 7-5
non-scan mode 4-5
rapid deployment F-1
RD bar codes E-11
scan/select key 3-2, 4-7, 4-8, 4-9, 4-17, 4-19
scannerdebuglevel 14-8, 14-9
scansamp app 4-19, 14-11
settings 4-28
specs A-2
supported code types C-2
troubleshooting 14-6, 14-7, 14-11
wireless 6-58
wireless scanning 1-2
scanning *See data capture*
scansamp app 4-19, 14-11
SCCP 4-37, 4-38, 4-39, 6-49,
. . . . . 6-50, 6-51, 7-9, 9-8,
. . . . . 9-9, 11-15, 14-23, 14-25
screen indications 4-12
search, server GUI 9-52
security
server passwords 6-18, 6-40
wireless protocols 4-14, A-2
server
capabilities when server is offline 6-58
demo solution 12-1, E-1
deployment example D-1

- firewalls 6-47
- functionality
  - device registration 6-4
  - login process 6-4
  - user profile delivery 6-4
  - voice communication setup
    - console for server setup 6-4
    - metrics collection 6-4
    - text commands 6-4
    - unicast calls 6-4
    - voice calls 6-4
    - walkie-talkie communication 6-4
- GUI 9-2
- installation
  - demo version 6-20
  - prerequisites 6-9
  - procedure 6-10
  - process 6-9
  - production version 6-11
  - software 6-8
  - upgrade demo to production 6-30
  - upgrading software revisions 6-31
- maintenance 14-2
- requirements 6-7
- security
- setup 6-7
- warranty iii
- Web Console
- WebConsole 9-2

service information xxi
settings
- backlight 4-27
- certificates 4-29
- date/time 4-29
- network 4-29
- scanner 4-28
- system properties 4-29

signal strength 4-5, 4-22
single slot USB serial cradle
- charging B-6

single-slot CA50 charging cradle B-5
single-slot cradle B-2
SIP 1-3, 4-37, 4-38, 4-40
sleep mode 3-9, 3-10, 3-13, 4-8, 4-18, 4-30, 10-6
soft keys 3-2, 4-7
software, server B-2
spare batteries 3-4, 3-8, B-2
spare battery charger B-2, B-5, B-8
- setup B-8

speaker 3-2, 4-9
staging & localization 12-6
standard calls 4-37
start menu 3-12, 4-4, 4-18
starting CA50 . . . . . 3-11, 4-2
static menu
- code sample 7-13
- modifying phone book menu 7-17
- overview 7-15
- phone book web service methods 7-19

status icons 4-5, 4-6
suspend 3-9, 3-10, 3-13, 4-18, 4-30, 10-6
symbologies
- enable/disable C-1

system info 4-29
system management
- client settings 9-40
- configure system settings 9-42
- high level system settings 9-42
- phone settings 9-41
- system settings 9-37

system properties settings 4-29
system settings
- Client.Backlight.Battery.Timeout 6-48
- Client.Backlight.Level 6-48
- Client.Desktop.Link.DefaultScan 6-48
- Client.NewPhoneRegRetryCount 6-48
- Client.Phone.Ring 6-48
- Client.PhoneRegRetryCount 6-48
- Client.PhoneRegRetryInterval 6-48
- Client.Volume.Receiver 6-48
- Client.Volume.Speaker 6-48
- Phone.Call.UnicastPacketRate 6-48
- Phone.Gateway.CallPark.ByGroup.Enable 6-48
- Phone.Gateway.CallPark.Enable 6-48
- Phone.Gateway.FAC.CallPark 6-48
- Phone.Gateway.FAC.CallPick 6-48
- Phone.Gateway.IPAddress 6-48
- Phone.Gateway.Port 6-49, 6-50, 6-51
- Phone.Gateway.Protocol.Type 6-49, 6-50, 6-51
- Phone.Gateway.Type 6-49, 6-50, 6-51
- Phone.Peer2Peer.Enable 6-49, 6-50, 6-51
- Phone.Peer2Peer.Port 6-49, 6-50, 6-51
- Phonebook.Ranges.BaseRangeCount 6-49, 6-50, 6-51
- Phonebook.Ranges.MaxNonRangeSize 6-49, 6-50, 6-51
- Phonebook.Ranges.MaxRangeSize 6-49, 6-50, 6-51
- Phonebook.RecentCalls.Interval 6-49, 6-50, 6-51
- Phonebook.SelectableGroups.ShowAll 6-49, 6-50, 6-51
- System.AllowDynamicWhenStaticFails 6-49, 6-50, 6-51
- System.DefaultLocation 6-49, 6-50, 6-51
- System.Device.ActivityThreshold . . 6-49, 6-50, 6-51

System.Language.LanguageID . . . 6-49, 6-50, 6-51
System.Logging.Level 6-49, 6-50, 6-51
System.Login.MACAuthOnly 6-52
System.Login.Token.GetPrimaryGroup 6-52
System.Login.Token.GetUserID 6-52
System.Login.Token.GetUserPW 6-52
System.Login.UserIDInputMethod 6-52
System.Maintenance.DB.CallParkExt.DaysBetween 6-52
System.Maintenance.DB.CallParkExt.FileAging 6-52
System.Maintenance.DB.CallParkExt.RecordAging 6-52
System.Maintenance.DB.DaysBetween 6-52
System.Maintenance.DB.DebugTrace.DaysBetween 6-52
System.Maintenance.DB.DebugTrace.FileAging 6-52
System.Maintenance.DB.DebugTrace.RecordAging 6-52
System.Maintenance.DB.Devices.DaysBetween 6-52
System.Maintenance.DB.Devices.RecordAging 6-52
System.Maintenance.DB.Export.Path 6-52
System.Maintenance.DB.Hour 6-52
System.Maintenance.DB.LocalUser.DaysBetween 6-52
System.Maintenance.DB.LocalUser.RecordAging 6-52
System.Maintenance.DB.Metrics.DaysBetween 6-52
System.Maintenance.DB.Metrics.FileAging . . . 6-52
System.Maintenance.DB.Metrics.RecordAging 6-52
System.Maintenance.Session.PollPeriod 6-53
System.Maintenance.Session.Timeout 6-53
System.Notif.CA50MediaRelay.Port 6-53
System.Notif.MCChannel.Address 6-53
System.Notif.MCChannel.Port 6-53
System.Notif.MediaRelay.DefaultAudioEncoding 6-53
System.Notif.MediaRelay.IPAddress 6-53
System.Notif.MediaRelay.Port 6-53
System.Profile.Path 6-53
System.Profile.Schema.Path 6-53
System.Profile.Template.Path 6-53
System.Profile.Workspace.Path 6-53
System.Resource.Path 6-53
System.Tools.Stage 6-53
System.UCAServer.IPAddress 6-53
System.Webservice.CallManager.URI 6-53
System.Webservice.CallPark.URI 6-53
WT.Base.MCAddress 6-53
WT.Base.Port 6-53
WT.ChannelIncrementMethod 6-53
system status
devices 9-6
server status 9-5

T

T_DebugTrace table 6-49, 6-50, 6-51
technical specifications A-1
accessories A-4
temperature, battery 3-7, 14-6
tools 9-47
troubleshooting
CA50 client 14-6
CA50 device diagnostic apps 14-11
log file 14-8
park/retrieve call
caller terminates park 4-52
calls parked by devices other than the CA50 4-52
parked call timeout 4-52
two devices attempt to pick up same parked call 4-52
server installation/operation 14-4
UCA server install/uninstall 14-4

U

UCA client app . . . . . 4-17, 4-19, 4-31, 4-36, ... 5-1, 6-6, 10-2, 12-2, 12-6
UCA server security *See server*
UCA server Web Console *See server*
UCA SysInfo app 4-5
UCA SysInfo app on CA50 device 4-22, 4-23
unicast call
and launching business apps 4-56
and walkie-talkie communication . . . 4-31, 4-34, 4-36
answering a voice call 4-45
browsing phone book 4-42
communication setup and maintenance 6-4
demo solution 6-5
functionality 6-58
indications 4-10, 4-11, 4-12
making a voice call 4-45
microphone 3-2, 4-9
mute mode 4-5
on call icon 4-5
overview 4-36
parking a call 4-47
receiver 4-9
retrieving a parked call 4-50
sound 4-9
speaker 3-2, 4-9

standard 4-37
states 4-53
technical spec A-2
transferring a call 4-46
usage 3-10
voice call functionality 1-3
WebConsole 9-2, 9-7
when scanning 4-54
uninstall
UCA server software 6-42
unpacking 3-1
up arrow key 3-2, 4-7
upgrading demo version UCA server software 6-30
upgrading software revisions 6-31
USB cable B-12
user management
methods
adddeviceuser 11-29
removedeviceuser 11-32
setdeviceuserpin 11-34
updatedeviceuser 11-33
user profile elements 7-4, 7-16
using earbud 4-13

V

voice call
answering a voice call 4-45
answering call while in a business app 4-55
browsing phone book 4-42
communication setup and maintenance 6-4
demo solution 6-5
functionality 1-3, 6-58
hold 4-46, 4-47, 4-50
indications 4-10, 4-11, 4-12
launching business apps while in call 4-56
making a call 4-45
microphone 3-2, 4-9
mute mode 4-5
on call icon 4-5
overview 4-36
parking 4-47
parking a call 4-47
receiver 4-9
retrieving a parked call 4-50
retrieving parked 4-50
sound 4-9
speaker 3-2, 4-9
standard 4-37
states 4-53
technical spec A-2
transferring a call 4-46
usage 3-10
WebConsole 9-2, 9-7
when scanning 4-54
voice quality 3-10
volume key 3-3, 4-7, 4-9

W

wake CA50 . . . . . 3-11, 4-2, 4-18, 14-6
walkie-talkie
and business apps 4-57
and unicast calls 4-36
and voice calls 4-36
calls 4-31
changing transmit channel 4-34
channels 4-33, 6-4, 6-6
communication 4-31, 6-2, 6-5
communication setup and maintenance 6-4
configuration 4-31
functions
channel settings 4-32, 7-4
indications 4-10, 4-11, 4-12
initiating connection 4-34
keys 3-2, 3-3, 4-7
modes
do not disturb 4-33, 7-5
normal 4-33, 7-5
scan 4-33, 7-5
single channel 4-33, 7-5
multicast functionality 1-3
naming channels 4-33, 7-24
receiving 4-34
sound 4-9
speaker 3-2, 4-9
WebConsole 9-2, 9-7
walkie-talkie profile elements 7-4
wall mount bracket B-2
wall mount brackets B-13, B-15
warm boot 4-8, 4-17, 10-7, 12-6, 14-6
warm boot (to access CA menu) 4-18
warranty
accessories iii
CA50 device iii
UCA server iii
Web Console 6-5
WebConsole
delete utility 9-52, 9-53
Device User Accounts 9-18
find utility 9-52
login 9-3
overview 9-2
password 9-10, 9-23
phone/walkie-talkie 9-7
Profile Management 9-26
Roles and Groups 9-30
System Management 9-37

System Status 9-5
tools 9-47
utilities
delete 9-52
find 9-52
import 9-55
wiping device 14-3

X

XML profile
alerts 7-9
audio 7-4
debuglog 7-12
desktop 7-7
device config 7-7
downloadfiles 7-6
menu states 7-12
metricslog 7-11
phone 7-4
ring tone 7-16
user 7-4
walkie-talkie 7-4
xml profile
alert settings 8-16
client settings 8-15
create menus 8-21
create new profile 8-4
desktop settings 8-11
device settings 8-14
download files settings 8-10
metrics & logging 8-13
phone settings 8-6
volume settings 8-12
walkie-talkie settings 8-7
XML sample
base XML profile 7-25
completed user XML profile 7-27
XML user profile 7-4

Tell Us What You Think...

We'd like to know what you think about this Manual. Please take a moment to fill out this questionnaire and fax this form to: (631) 627-7184, or mail to:

Motorola
One Motorola Plaza M/S B-10
Holtsville, NY 11742-1300
Attention: Technical Publications Manager

IMPORTANT: If you need product support, please call the appropriate customer support number provided. Unfortunately, we cannot provide customer support at the fax number above.

Manual Title:_____
(please include revision level)

How familiar were you with this product before using this manual?

☐ Very familiar ☐ Slightly familiar ☐ Not at all familiar

Did this manual meet your needs? If not, please explain.

_____
_____
_____
_____

What topics need to be added to the index, if applicable?

_____
_____
_____
_____

What topics do you feel need to be better discussed? Please be specific.

_____
_____
_____
_____

What can we do to further improve our manuals?

_____
_____
_____
_____

Thank you for your input—We value your comments.

Quick Startup Guide

Introduction to CA50 Solution *1-1, 1-2*

Walkie-Talkie *3-2, **4-31**, 4-34, 4-56, 5-1, 6-4, 9-7, 13-7, 14-37, E-1*

- Channel Selection; Naming *4-34, 7-24*

Voice Calls *5-1, 6-4, 9-14, 9-7, 14-18, 14-17, **4-36**, E-1*

- Phone Book ..... *13-7, 13-10 , 4-36, **4-42***
- Making/Receiving/Transfering Calls ***4-45**, 4-46*

Business Applications Like Price Lookup *4-54, 6-5, **11-2**, 13-10, E-1*

Messaging/Paging ***1-2**, 11-26, 11-44*

Components in CA50 Solution *1-4*

A Good Place to Start *1-1, 1-2*

Development and Deployment Checklist ..... *2-1, **2-2** , 13-2*

Self Contained Pre-Built Demo *E-1*

Deployment Example: UCA Server Install and Basic User Setup *D-1*

CA50 - Hardware

Physical Key Layout ***3-2**, 4-7, 4-19*

Volume Control/Mute Modes/Headset *3-2, 4-7, 4-13*

CA50 - Configuration/Staging/Deploying

Reboot Device/Access the CA Menu *3-12, 4-17, **4-18***

Device Staging and Deployment using MSP *3-12, 4-19, **12-1**, 12-2, D-16, F-1*

XML Profile Configures CA50 when Downloaded from Server ***7-1**, D-6*

Battery Management Recommendations *3-6, **3-9**, 4-4, A-1, B-8, B-12*

Determining Version Information (OS, Client, etc.) *4-2, **4-29***

Server *5-1, **6-2**, D-1*

Installing/Uninstalling Server Software ***6-10**, 6-42, 6-45*

- Server Hardware / Software Requirements *6-7*
- Validate Successful Server Installation and Proper Performance *6-45*

Web Console for Server Setup and Maintenance *9-2, 14-2*

- Login and Password ***9-4**, 6-13*
- Populate Database with User/Site Specific Information *2-3*
- CA50 Capabilities with UCA Server *6-58*
- Importing Users and Extensions ***9-13**, 9-25*
- Troubleshooting/Error Codes ***14-4**, 14-6, 14-18*
- Alternate Administrative Mechanism to Manage Customer Data ***11-5**, 11-14, 11-18, 11-22, 11-29, 11-36*

Business Applications *5-1, **10-2**, 10-2, 10-12, E-1*

Launching Business Applications *10-2, 12-2*

Employee Identity Validation: Login Bar Codes and PIN ***4-3**, 11-5, 11-40, D-15, E-14*

HTML Elements *10-5*

Troubleshooting/Error Codes. ***14-4**, 14-6, 14-13*

PBX Integration (H.323, SIP, SCCP) ***4-37**, 4-38, 6-48, 9-13, 9-41, 13-10*

MOTOROLA

Motorola, Inc.
One Motorola Plaza
Holtsville, New York 11742, USA
1-800-927-9626
http://www.motorola.com/enterprisemobility

MOTOROLA and the Stylized M Logo and Symbol and the Symbol logo are registered in the U.S. Patent and Trademark Office.
All other product or service names are the property of their registered owners.
© Motorola, Inc. 2009

72E-93914-02 Revision A - November 2009